



Junta de Andalucía

Guía de referencia de JTrApi

Trew@ v2.6.10.1

Versión: v01r00
Fecha: 03/03/2025

Queda prohibido cualquier tipo de explotación y, en particular, la reproducción, distribución, comunicación pública y/o transformación, total o parcial, por cualquier medio, de este documento sin el previo consentimiento expreso y por escrito de la Junta de Andalucía.

	<p>Junta de Andalucía Agencia Digital de Andalucía</p>	<p>Trew@ Guía de referencia de JTrApi</p>
---	---	---

HOJA DE CONTROL

Título	Guía de referencia de JTrApi		
Entregable	Trew@		
Nombre del Fichero	TRW26101E_MUS_ Guía referencia JTRAPI_v01r00.odt		
Autor	UTE		
Versión/Edición	v01r00	Fecha Versión	03/03/2025
Aprobado por	ADA	Fecha Aprobación	
		Nº Total Páginas	766

REGISTRO DE CAMBIOS

Versión	Causa del cambio	Responsable del cambio	Área	Fecha del cambio
v01r00	Versión inicial	UTE	UTE	03/03/2025

CONTROL DE DISTRIBUCIÓN

Nombre y Apellidos	Cargo	Área
Juan Luis Ceada Ramos	Subdirector	Servicios Digitales y Analítica de Datos
Manuel Escobar Montes	Jefe de Servicio	Administración Digital
José Antonio Frutos Beltrán	Técnico	Gabinete Estratégico
Almudena López Maraver	Directora de Proyecto	PMO

	<p>Junta de Andalucía Agencia Digital de Andalucía</p>	<p>Trew@ Guía de referencia de JTrApi</p>
---	---	---

Nombre y Apellidos	Cargo	Área
José Luis Romero Dorado	Jefe de proyecto	UTE
Juan Carlos Gómez de Tejada Díaz	Consultor	UTE

ÍNDICE

1 INTRODUCCIÓN.....	10
1.1 Objeto.....	10
1.2 Alcance.....	10
2 CONTENIDO DEL DOCUMENTO.....	11
2.1 Introducción.....	11
2.2 Una visión global de las J-TrAPIs.....	11
2.2.1 Generalidades.....	12
2.2.1.1 Datos devueltos.....	12
2.2.1.2 Filtros y orden sobre los datos.....	12
2.2.1.3 Errores.....	13
2.2.1.4 Estructura de paquetes.....	13
2.2.1.5 Apis integradas con Alfresco.....	14
2.2.1.6 Apis que se comunican con el bus de w@ndA.....	16
2.2.1.7 Clases SQL. El paquete trewa.bd.sql.....	16
2.2.2 Campos y valores.....	17
2.2.2.1 Campo.....	17
2.2.2.2 TipoCampo.....	17
2.2.3 Operadores.....	18
2.2.3.1 OperadorWhere.....	18
2.2.3.2 OperadorOrderBy.....	18
2.2.3.3 OperadorLogico.....	19
2.2.4 Expresiones.....	19
2.2.4.1 ExpresionWhere.....	19
2.2.4.2 ExpresionOrderBy.....	20
2.2.5 Cláusulas.....	20
2.2.5.1 ClausulaWhere.....	20
2.2.5.2 ClausulaOrderBy.....	22
2.2.5.3 ClausulaExclusion.....	23
2.2.6 Clases de Excepción.....	24
2.2.6.1 TrException.....	24
2.2.6.2 TrExceptionMensajeCondicion.....	25
2.2.7 Calendarios hábiles.....	26
2.2.8 Selector de plantillas múltiples.....	27
2.3 Datos de auditoría.....	28
2.4 JTrApis.....	28
2.4.1 Descripción de la TrAPIUI.....	28
2.4.1.1 Descripción de clases involucradas.....	28
2.4.1.1.1 Interfaz J-TrAPIUI.....	28
2.4.1.1.2 Clase Factoría.....	29
2.4.1.2 Clases de entidad.....	33
2.4.1.2.1 TrAccion.....	33



2.4.1.2.2 TrAccionDocumento.....	34
2.4.1.2.3 TrAccionDocumentoPortafirmas.....	34
2.4.1.2.4 TrAccionTransicion.....	35
2.4.1.2.5 TrAvisoCaducidad.....	36
2.4.1.2.6 TrBloque.....	36
2.4.1.2.7 TrBloqueExplorador.....	38
2.4.1.2.8 TrBloquePermitido.....	39
2.4.1.2.9 TrCaducidad.....	40
2.4.1.2.10 TrCaducidadExpediente.....	41
2.4.1.2.11 TrCambioProcedimientoExpediente.....	42
2.4.1.2.12 TrComponente.....	43
2.4.1.2.13 TrComunicacion.....	44
2.4.1.2.14 TrCondicion.....	45
2.4.1.2.15 TrCondicionAccionAviso.....	45
2.4.1.2.16 TrCondicionDocumento.....	46
2.4.1.2.17 TrCondicionTransicion.....	47
2.4.1.2.18 TrConstante.....	48
2.4.1.2.19 TrConstanteGn.....	49
2.4.1.2.20 TrConsultaCertificado.....	49
2.4.1.2.21 TrDatoComponente.....	50
2.4.1.2.22 TrDatosContacto.....	51
2.4.1.2.23 TrDatosRegistroDocumentos.....	53
2.4.1.2.24 TrDatoTipoPago.....	54
2.4.1.2.25 TrDiaFestivo.....	54
2.4.1.2.26 TrDefProcedimiento.....	55
2.4.1.2.27 TrDocMultiple.....	58
2.4.1.2.28 TrDocumentoExpediente.....	58
2.4.1.2.29 TrDocumentoExpedienteSimple.....	65
2.4.1.2.30 TrDocumentoExplorador.....	68
2.4.1.2.31 TrDocumentoPermitido.....	69
2.4.1.2.32 TrEmpleado.....	71
2.4.1.2.33 TrEnviarA.....	73
2.4.1.2.34 TrEvolucionExpediente.....	73
2.4.1.2.35 TrExpediente.....	75
2.4.1.2.36 TrExpedienteCaducado.....	78
2.4.1.2.37 TrExpedienteSimple.....	81
2.4.1.2.38 TrExplorador.....	83
2.4.1.2.39 TrFamiliaSubfamilia.....	87
2.4.1.2.40 TrFase.....	88
2.4.1.2.41 TrFaseActualExpediente.....	89
2.4.1.2.42 TrFaseExpediente.....	91
2.4.1.2.43 TrFirmaDocumentoExpediente.....	93
2.4.1.2.44 TrFirmante.....	95
2.4.1.2.45 TrFirmanteTipoDocumento.....	96
2.4.1.2.46 TrIndicacionFicha.....	97
2.4.1.2.47 TrInteresado.....	98
2.4.1.2.48 TrInteresadoSimple.....	100
.....	101
2.4.1.2.49 TrInteresadoDocumento.....	101
2.4.1.2.50 TrInteresadoExpediente.....	103
2.4.1.2.51 TrLimiteRelacion.....	105
2.4.1.2.52 TrMensaje.....	106
2.4.1.2.53 TrMensajeCondicionAccion.....	108



.....	108
2.4.1.2.54 TrMetadato.....	108
2.4.1.2.55 TrMetafase.....	111
2.4.1.2.56 TrModificacionCaducidadExpediente.....	113
2.4.1.2.57 TrMunicipio.....	114
.....	114
2.4.1.2.58 TrNormativaProcedimiento.....	115
2.4.1.2.59 TrNotificacionEnvio.....	115
2.4.1.2.60 TrOrganismo.....	117
2.4.1.2.61 TrOrganismoExpediente.....	120
2.4.1.2.62 TrOrganismoProcedimiento.....	121
2.4.1.2.63 TrPagoTasa.....	123
2.4.1.2.64 TrPais.....	124
2.4.1.2.65 TrParametro.....	124
2.4.1.2.66 TrParametroBloque.....	125
2.4.1.2.67 TrParrafo.....	125
2.4.1.2.68 TrParrafoTipoDocumento.....	127
2.4.1.2.69 TrPerfilUsuario.....	129
2.4.1.2.70 TrPeticionEliminada.....	130
2.4.1.2.71 TrPlantillaProcedimiento.....	130
2.4.1.2.72 TrPlazo.....	131
2.4.1.2.73 TrProvincia.....	131
2.4.1.2.74 TrPtoTrabOrganismo.....	132
2.4.1.2.75 TrPuestoTrabajo.....	133
2.4.1.2.76 TrRazonInteres.....	134
.....	134
2.4.1.2.77 TrRazonInteresDocumento.....	135
2.4.1.2.78 TrRazonInteresProcedimiento.....	136
2.4.1.2.79 TrRegistroOperacion.....	137
2.4.1.2.80 TrRelacionDefinida.....	137
2.4.1.2.81 TrRelacionExpediente.....	138
2.4.1.2.82 TrRelacionInteresado.....	139
2.4.1.2.83 TrSistema.....	141
2.4.1.2.84 TrTareaExpediente.....	144
2.4.1.2.85 TrTareaPermitida.....	146
2.4.1.2.86 TrTextoDisposicion.....	148
2.4.1.2.87 TrTipoActo.....	149
2.4.1.2.88 TrTipoCertificado.....	150
2.4.1.2.89 TrTipoCertificadoProcedimiento.....	151
2.4.1.2.90 TrTipoContacto.....	151
2.4.1.2.91 TrTipoDocumento.....	152
2.4.1.2.92 TrTipoExpediente.....	157
2.4.1.2.93 TrTipoidentificador.....	158
2.4.1.2.94 TrTipoIndicacion.....	159
2.4.1.2.95 TrTipoOrganismo.....	160
2.4.1.2.96 TrTipoOrganizacion.....	161
2.4.1.2.97 TrTipoPago.....	161
2.4.1.2.98 TrTipoPagoProcedimiento.....	163
2.4.1.2.99 TrTipoParrafo.....	164
2.4.1.2.100 TrTipoRelacion.....	165
2.4.1.2.101 TrTipoVia.....	166
2.4.1.2.102 TrTransicion.....	167
2.4.1.2.103 TrTransicionDefProcedimiento.....	169



2.4.1.2.104	TrTransicionExplorador.....	170
2.4.1.2.105	TrUsuario.....	171
2.4.1.2.106	TrUsuarioAsignado.....	172
2.4.1.2.107	TrUsuarioAsignadoExplorador.....	174
2.4.1.2.108	TrUsuarioPerfilUsuario.....	174
2.4.1.2.109	TrValorParametro.....	176
2.4.1.2.110	TrVariable.....	177
2.4.1.2.111	TrVariableDocExp.....	178
2.4.1.2.112	TrVersionDefProcedimiento.....	178
	179
2.4.1.2.113	TrIndiceElectronico.....	179
2.4.1.2.114	TrIdentificadorEni.....	180
2.4.1.3	<i>Servlets de subida y descarga de documentos.....</i>	<i>181</i>
2.4.1.3.1	Servlet DescargaDocumento.....	181
2.4.1.3.2	Servlet SubidaDocumento.....	182
2.4.1.4	<i>Métodos de la TrAPIUI.....</i>	<i>183</i>
2.4.1.4.1	APIs sobre documentos.....	183
2.4.1.4.2	APIs sobre firmas de documentos.....	271
2.4.1.4.3	APIs exclusivas para Expediente Electrónico.....	282
2.4.1.4.4	APIs sobre expedientes.....	292
2.4.1.4.5	APIs sobre procedimientos definidos.....	323
2.4.1.4.6	APIs sobre tramitación (transiciones y fases).....	330
2.4.1.4.7	APIs sobre bloques de datos (tareas II).....	349
2.4.1.4.8	APIs sobre caducidades.....	360
2.4.1.4.9	APIs sobre el sistema de intercambio de mensajes.....	366
2.4.1.4.10	APIs sobre interesados.....	371
2.4.1.4.11	Otras APIs.....	386
2.4.1.4.12	APIs exclusivas para Port@firmas.....	387
2.4.1.4.13	APIs exclusivas para @visor.....	389
2.4.1.4.14	APIs Exclusivas para Notific@.....	392
2.4.1.4.15	Otras APIs de obtención de datos definidas.....	395
2.4.2	Descripción de la TrAPIADM.....	431
2.4.2.1	<i>Descripción de clases involucradas.....</i>	<i>431</i>
2.4.2.1.1	Interfaz J-TrAPIADM.....	431
2.4.2.1.2	Clase Factoría.....	431
2.4.2.1.3	Clases de entidad.....	434
2.4.2.2	<i>Métodos de la TrAPIADM.....</i>	<i>533</i>
2.4.2.2.1	Mantenimiento de Certificados.....	533
2.4.2.2.2	Mantenimiento de metafases.....	534
2.4.2.2.3	Mantenimiento de Tipo certificados SCSP.....	537
2.4.2.2.4	Mantenimiento de definiciones de procedimientos.....	539
2.4.2.2.5	Mantenimiento de documentos.....	543
2.4.2.2.6	Mantenimiento de expedientes.....	543
2.4.2.2.7	Mantenimiento de fases.....	546
2.4.2.2.8	Mantenimiento de transiciones.....	548
2.4.2.2.9	Mantenimiento de perfiles de usuario.....	551
2.4.2.2.10	Mantenimiento de tipos de expediente.....	552
2.4.2.2.11	Mantenimiento de versiones de procedimientos.....	555
2.4.2.2.12	Mantenimiento de extremos de transiciones gráficas.....	557
2.4.2.2.13	Mantenimiento de definición de procedimientos gráficos.....	559
2.4.2.2.14	Mantenimiento de metafases gráficas.....	561
2.4.2.2.15	Mantenimiento de nodos de transición gráfica.....	562
2.4.2.2.16	Mantenimiento de transiciones gráficas.....	564



2.4.2.2.17	Mantenimiento de tareas (Manipulación datos y otras).....	566
2.4.2.2.18	Mantenimiento de tareas en fase (Manipulación datos y otras) .	567
2.4.2.2.19	Mantenimiento de caducidades.....	569
2.4.2.2.20	Mantenimiento de tareas (Manipulación de escritos).....	571
2.4.2.2.21	Mantenimiento de tareas en fase (Manipulación de escritos).....	573
2.4.2.2.22	Mantenimiento de condiciones.....	575
2.4.2.2.23	Mantenimiento de acciones.....	577
2.4.2.2.24	Mantenimiento de avisos.....	579
2.4.2.2.25	Mantenimiento de condiciones de una transición.....	581
2.4.2.2.26	Mantenimiento de acciones de una transición.....	584
2.4.2.2.27	Mantenimiento de avisos de una transición.....	587
2.4.2.2.28	Mantenimiento de condiciones de documentos.....	590
2.4.2.2.29	Mantenimiento de acciones de documentos.....	593
2.4.2.2.30	Mantenimiento de metadatos.....	596
2.4.2.2.31	Mantenimiento de avisos de documentos.....	598
2.4.2.2.32	Mantenimiento de condiciones de otras tareas.....	601
2.4.2.2.33	Mantenimiento de acciones de otras tareas.....	603
2.4.2.2.34	Mantenimiento de avisos de otras tareas.....	605
2.4.2.2.35	Mantenimiento de datos de un componente.....	607
2.4.2.2.36	Mantenimiento de componentes.....	608
2.4.2.2.37	Mantenimiento de constantes del sistema.....	610
2.4.2.2.38	Mantenimiento de tipos de actos.....	611
2.4.2.2.39	Mantenimiento de perfiles de usuario de una transición.....	613
2.4.2.2.40	Mantenimiento de límites de caducidades.....	615
2.4.2.2.41	Mantenimiento de documentos delegados.....	617
2.4.2.2.42	Mantenimiento de ficha del procedimiento.....	619
2.4.2.2.43	Mantenimiento de normativas.....	621
2.4.2.2.44	Mantenimiento de normativas del procedimiento.....	623
2.4.2.2.45	Mantenimiento de organismos.....	625
2.4.2.2.46	Mantenimiento de organismos del procedimiento.....	627
2.4.2.2.47	Mantenimiento de plantillas del procedimiento.....	629
2.4.2.2.48	Mantenimiento de tipo certificado del procedimiento.....	631
2.4.2.2.49	Mantenimiento de errores.....	633
2.4.2.2.50	Mantenimiento de firmantes definidos.....	635
2.4.2.2.51	Mantenimiento de firmas.....	637
2.4.2.2.52	Mantenimiento de plantillas.....	639
2.4.2.2.53	Mantenimiento de relaciones.....	641
2.4.2.2.54	Mantenimiento de disposiciones de firma.....	643
2.4.2.2.55	Mantenimiento de parámetros.....	645
2.4.2.2.56	Mantenimiento de párrafos.....	647
2.4.2.2.57	Mantenimiento de tipos de párrafos.....	649
2.4.2.2.58	Mantenimiento de tipos de relaciones.....	651
2.4.2.2.59	Mantenimiento de variables.....	653
2.4.2.2.60	Mantenimiento de perfiles de usuario de otras tareas.....	655
2.4.2.2.61	Mantenimiento de perfiles de usuario de documentos.....	657
2.4.2.2.62	Mantenimiento de firmas de tipos de documentos.....	660
2.4.2.2.63	Mantenimiento de parámetros de otras tareas.....	662
2.4.2.2.64	Mantenimiento de parámetros de variables.....	664
2.4.2.2.65	Mantenimiento de perfiles de usuario de un usuario.....	666
2.4.2.2.66	Mantenimiento de ámbitos de ley.....	668
2.4.2.2.67	Mantenimiento de razones de interés.....	670
2.4.2.2.68	Mantenimiento de tipos de pago.....	672
2.4.2.2.69	Mantenimiento de tipos de componentes.....	674



2.4.2.2.70	Mantenimiento de tipos de contacto.....	676
2.4.2.2.71	Mantenimiento de tipos de indicaciones.....	678
2.4.2.2.72	Mantenimiento de tipos de normativas.....	680
2.4.2.2.73	Mantenimiento de tipos de publicación.....	682
2.4.2.2.74	Mantenimiento de variables para un tipo de documento.....	684
2.4.2.2.75	Mantenimiento de constantes generales.....	686
2.4.2.2.76	Mantenimiento de empleados.....	688
2.4.2.2.77	Mantenimiento de provincias.....	691
2.4.2.2.78	Mantenimiento de municipios.....	693
2.4.2.2.79	Mantenimiento de países.....	695
2.4.2.2.80	Mantenimiento de puestos de trabajo.....	697
2.4.2.2.81	Mantenimiento de puestos de trabajo de un organismo.....	699
2.4.2.2.82	Mantenimiento de sistemas.....	701
2.4.2.2.83	Mantenimiento de tipos de identificadores.....	703
2.4.2.2.84	Mantenimiento de tipos de organismos.....	705
2.4.2.2.85	Mantenimiento de tipos de organizaciones.....	707
2.4.2.2.86	Mantenimiento de tipos de vías.....	709
2.4.2.2.87	Mantenimiento de usuarios.....	711
2.4.2.2.88	Mantenimiento de días festivos.....	714
2.4.2.2.89	Otras APIs.....	717
2.4.2.2.90	APIs exclusivas para Expediente Electrónico.....	720
2.4.3	Descripción del TrAPIUTL.....	722
2.4.3.1	Descripción de clases involucradas.....	722
2.4.3.1.1	Interfaz J-TrAPIUTL.....	722
2.4.3.1.2	Clase Factoría.....	728
2.4.3.1.3	Clases de entidad.....	730
2.4.3.1.4	Servlets de subida y descarga de XML (DDP, Expedientes y Entidades del sistema).....	731
2.4.3.1.5	Servlet DescargaXML.....	731
2.4.3.1.6	Servlet SubidaXML.....	733
2.4.3.1.7	Servlet SubidaXML en formato ZIP.....	734
2.4.3.1.8	Servlet JustificanteBorrado.....	736
2.4.4	Despliegue y configuración de las J-TrAPIs.....	737
2.4.4.1	Integración con Maven.....	737
2.4.4.2	Despliegue de las J-TrAPIs.....	737
2.4.4.3	Configuración de servlets.....	737
2.4.4.4	Configuración de perfiles de conexión.....	740
2.4.4.5	Configuración con archivo de properties.....	740
2.4.4.6	Formato del fichero de properties.....	741
2.4.4.7	Configuración generación documentos OpenOffice.....	744
2.4.4.8	Trazabilidad con log4java.....	745
2.4.4.9	Configuración para WebOffice.....	746
2.4.4.10	Despliegue de los Servicios Web de Trew@.....	747
2.4.4.11	Condiciones, acciones, variables y selectores de plantillas Java.....	748
3	BIBLIOGRAFÍA Y REFERENCIAS.....	751

	Junta de Andalucía Agencia Digital de Andalucía	Trew@ Guía de referencia de JTrApi
---	---	---------------------------------------

1 INTRODUCCIÓN

1.1 Objeto

El presente documento recoge el manual de referencia de la JTrApi de Trew@ v2.6.10.1

1.2 Alcance

Este documento va dirigido a:

- La dirección del proyecto w@ndA.
- El colectivo de usuarios y desarrolladores de sistemas que usan Trew@ como motor de tramitación.

2 CONTENIDO DEL DOCUMENTO

2.1 Introducción

TREW@ como tramitador de procedimientos, dispone de una serie de funcionalidades mediante las cuáles un sistema externo puede, en general:

- Gestionar la tramitación de expedientes conforme a la definición de los procedimientos que en Trew@ se han implementado
- Obtener todos los datos que sobre dicha tramitación se recoge en Trew@ a modo de “historia” de tramitación
- Obtener otro tipo de información acerca de los procedimientos definidos

Esta funcionalidad se ofrece a los sistemas que utilizan TREW@ mediante un conjunto de métodos, en adelante **J-TrAPIs**, las cuáles permiten interactuar con el metamodelo de datos que da soporte a Trew@.

2.2 Una visión global de las J-TrAPIs

Las **J-TrAPIs** son el punto de acceso de las aplicaciones al motor de tramitación Trew@. El objetivo de las mismas es abstraer a las aplicaciones del complejo modelo de datos y aportar una serie de funcionalidades enfocadas a tener un uso común de acceso para todas las aplicaciones.

Se han definido una serie de APIs según su funcionalidad:

- Interfaz de acceso UI – TrAPIUI

API para la gestión de operaciones de tramitación.

- Interfaz de acceso ADM – TrAPIADM

API para la gestión de operaciones de definición de procedimientos.

- Interfaz de acceso UTL – TrAPIUTL

API para el intercambio de definiciones de procedimiento entre diferentes sistemas.

2.2.1 Generalidades

El conjunto de APIs que ofrece Trew@ se encuentran implementadas en forma de métodos que acceden al modelo de datos para realizar las operaciones necesarias.

2.2.1.1 Datos devueltos

Un gran número de las APIs que existen tratan de devolver en algunos de sus parámetros información en forma de un conjunto de datos. Para ello, se aportan una serie de clases de entidad a través de las cuales se producirá el intercambio de información entre aplicaciones y el propio motor de Trew@. Se recomienda comprobar si se han obtenido datos preguntando si el array es distinto de null y su tamaño es mayor a cero. Por ejemplo:

```
TrFase[] datosFase = apiUI.obtenerDatosFase(new TpoPK(123),null,null);
if(datosFase!=null && datosFase.length > 0)
{
    //hay datos
    ...
}
```

En general para saber los datos devueltos por cualquier api se recomienda comprobar los campos por los cuáles permite filtrar y/o ordenar el api ya que los datos devueltos coinciden con éstos últimos.

2.2.1.2 Filtros y orden sobre los datos

La mayoría de los métodos de las APIs que devuelven un array de elementos permiten hacer filtros y ordenaciones sobre los mismos según se necesite. Esto se hace a través del paso de parámetro de objetos de 2 clases implementadas para montar sentencias Where y OrderBy: ClausulaWhere y ClausulaOrderBy. En muchos casos aparecerán sentencias del tipo:

```
Nombre_API_TREW@ (... , ClausulaWhere where, ClausulaOrderBy orderby )
```

En general para todas aquellas consultas sobre las que no se quiera aplicar filtros u ordenación, bastará con pasar "null" en el paso de estos parámetros.

Para una descripción completa de estas clases y su forma de uso, así como de las clases involucradas, se remite al apartado: [Clases SQL](#).

2.2.1.3 Errores

En general, los errores devueltos por Trew@ se harán en forma de excepciones que deberán ser capturadas por las aplicaciones para conseguir el mensaje de error.

Para una descripción completa de las clases de Excepción, se remite al apartado: [Clases de Excepción](#).

2.2.1.4 Estructura de paquetes

Las TrAPIs presentan una estructura organizada en paquetes según la funcionalidad expuesta en los mismos. La siguiente tabla describe los principales paquetes en los que se descompone la librería:

paquete	Descripción
<i>trewa</i>	Paquete principal del que cuelga toda la implementación de las J-TrAPIS.
<i>trewa.bd</i>	Paquete para clases de acceso a base de datos.
<i>trewa.bd.sql</i>	Paquete para clases SQL.
<i>trewa.bd.tpo</i>	Paquete para clases de enmascaramiento de tipos.
<i>trewa.bd.trapi</i>	Paquete principal de acceso a las J-TrAPIS.
<i>trewa.bd.trapi.trapiui</i>	Paquete para clases de acceso a la interfaz TrAPIUI.
<i>trewa.bd.trapi.trapiui.tpo</i>	Paquete para clases de entidad asociadas a la interfaz TrAPIUI.
<i>trewa.bd.trapi.trapiadm</i>	Paquete para clases de acceso a la interfaz TrAPIADM.
<i>trewa.bd.trapi.trapiutl</i>	Paquete para clases de acceso a la interfaz TrAPIUTL.
<i>trewa.conf</i>	Paquete para clases de configuración de Trew@.
<i>trewa.conf.perfiles</i>	Paquete para la especificación de perfiles.
<i>trewa.exception</i>	Paquete para clases de excepciones Trew@.
<i>trewa.ext</i>	Paquete para clases de extensión Trew@.
<i>trewa.util</i>	Paquete para clases de utilidad.

2.2.1.5 Apis integradas con Alfresco

Las apis de la interfaz TrAPIUI, marcadas en esta Guía de referencia con una **(w)** intercambian información de los documentos con el componente ContentManager, siempre y cuando se haya definido dicho componente en el sistema con el que estemos trabajando.

El componente se debe definir con los siguientes datos:

Dirección IP	Nombre de la máquina o dirección IP en la que se encuentra Alfresco	Por ejemplo: alfresco.guadaltel.es
Usuario	Usuario para autenticación con Alfresco	Por ejemplo: admin
Password	Clave para el usuario de autenticación con Alfresco	Por ejemplo: admin

A este componente se le deben incluir los siguientes datos de componente:

Dato componente	Descripción	Valores de ejemplo
<i>IMPL</i>	Nombre de la implementación del conector ContentManager	alfrescows
<i>CLASS</i>	Clase de la implementación indicada	trewa.contentmanager.impl. .AlfrescoWSImpl
<i>PROTOCOLO</i>	Protocolo para la comunicación con los servicios web de Alfresco	http
<i>PUERTO</i>	Puerto para la comunicación con los servicios web de Alfresco	8080
<i>ROOTSPACE</i>	Root space usado en Alfresco	/app:company_home/cm:TREWA
<i>STRATEGYIMPL</i>	Clase que implementa la estrategia que nos permite definir el funcionamiento de los métodos sabe, update, remove y get de la implementación de Alfresco.	trewa.contentmanager.strat egy.DefaultStrategyImpl
<i>INCLUDE_ROOTNAME</i>	Indica si se crea un directorio con el nombre del sistema en Alfresco para almacenar los documentos.	TRUE o FALSE
<i>INCLUDE_PRODUCTION _UNIT</i>	Indica si se crea un directorio con el código w@ndA del organismo en Alfresco.	TRUE o FALSE
<i>INCLUDE_PROCEDURE _TYPE</i>	Indica si se crea un directorio con el código w@ndA del procedimiento en Alfresco.	TRUE o FALSE
<i>INCLUDE_YEAR</i>	Indica si se crea un directorio con el año para almacenar los documentos.	TRUE o FALSE
<i>INCLUDE_MONTH</i>	Indica si se crea un directorio con el mes para almacenar los documentos.	TRUE o FALSE

Una vez definido el componente correctamente se debe asociar al sistema que queramos que envíe los documentos a Alfresco mediante el atributo "Guardar docs en..." seleccionando el nuevo componente creado.

Algunos datos de Trew@ que se envían al componente Alfresco son:

- El código w@ndA del componente.
- El ciwa del organismo al que pertenece el expediente.
- El código w@ndA del procedimiento.
- El código w@ndA del tipo de documento.

- El código w@ndA de los interesados y de sus razones de interés.

Una de las opciones respecto a la comunicación con gestores documentales es la posibilidad de usar exclusivamente el repositorio externo para guardar los documentos físicos, evitando de esta forma almacenarlos también en Trew@. Para ello debemos definir en nuestro sistema la constante "REPOSITORIO_DOC_EXT", cuyos valores pueden ser "S" (Sólo se usará el gestor documental externo para almacenar los documentos físicos) ó "N" (Los documentos se guardan en Trew@ y en el gestor documental externo si lo hubiese definido). Su valor por defecto es "N".

2.2.1.6 Apis que se comunican con el bus de w@ndA

Las apis de la interfaz TrAPIUI, marcadas en esta Guía de referencia con una **(b)** intercambian información de los expedientes con el bus w@ndA siempre y cuando se haya definido un componente bus con los datos necesarios asociado al sistema (apartado "Informar a...") y se haya marcado la conexión con dicho componente como "Sí" u "Opcional" (para más detalles ver la "Guía de integración w@ndA").

2.2.1.7 Clases SQL. El paquete trewa.bd.sql

Las clases SQL se usan para realizar filtros y ordenaciones en las peticiones a las TrAPIs aportando un mecanismo para montar cláusulas Where y Order By. Todas estas clases pertenecen al paquete **trewa.bd.sql**.

Básicamente una **cláusula Where** aparece implementada en la librería como un conjunto de "**Expresiones Where**" y otras "**cláusulas Where**" enlazadas unas con otras a través de "**Operadores lógicos**".

→ Una expresión Where presenta el siguiente formato:

< *ExpresionWhere* > = < *Campo* > + < *OperadorWhere* > + < *Valor* >

→ Una ClausulaWhere presenta alguno de los siguientes formatos:

< *ClausulaWhere* > = < *ExpresionWhere* | *ClausulaWhere* > + < *OperadorLogico* > + < *ExpresionWhere* | *ClausulaWhere* >

La **cláusula "Order By"** aparece implementada como un conjunto de "Expresiones Order By".

→ Una ClausulaOrderBy presenta el siguiente formato:

< *ClausulaOrderBy* > = < *ExpresionOrderBy* > + < *ExpresionOrderBy* > + ...

→ Una expresión OrderBy presenta el siguiente formato:

< *ExpresionOrderBy* > = < *Campo* > + < *OperadorOrderBy* >

2.2.2 Campos y valores

Un Campo para la librería es una columna sobre la que se va a poder filtrar u ordenar.

Los Campos son usados tanto por las expresiones Where como por las OrderBy y se obtendrán a través de las clases de entidad, para las cuales se han definido una serie de campos susceptibles de ser usados en los filtros y ordenaciones. Para ver una relación de los campos disponibles y los posibles valores de uso, consultar el apartado *Clases de Entidad*.

2.2.2.1 Campo

trewa.bd.Campo

Interfaz para aportar información sobre un campo.

Métodos

- String getNombreCampo()
Obtiene el nombre de un campo

- TipoCampo getTipo()
Obtiene el tipo de campo

2.2.2.2 TipoCampo

trewa.bd.TipoCampo

Clase que representa un tipo de campo.

Tipo de campo	Tipo
TIPO_NUMBER	"NUMBER"
TIPO_VARCHAR2	"VARCHAR2"
TIPO_DATE	"DATE"
TIPO_BOOLEAN	"BOOLEAN"

2.2.3 Operadores

Los operadores intervienen en las diferentes expresiones Where y OrderBy

2.2.3.1 OperadorWhere

trewa.bd.sql.OperadorWhere

Esta clase define un operador válido para una expresión where. Los posibles operadores habilitados son:

OperadorWhere	Operador
OP_LIKE	"LIKE"
OP_NOT_LIKE	"NOT LIKE"
OP_IGUAL	"="
OP_DISTINTO	"!="
OP_IS_NULL	"IS NULL"
OP_IS_NOT_NULL	"IS NOT NULL"
OP_MAYOR	">"
OP_MENOR	"<"
OP_MAYOR_IGUAL	">="
OP_MENOR_IGUAL	"<="
OP_IN	"IN"
OP_NOT_IN	"NOT IN"
OP_LIKE_IGNORE_CASE	"LIKE" No es sensible a mayúsculas y minúsculas
OP_NOT_LIKE_IGNORE_CASE	"NOT LIKE" No es sensible a mayúsculas y minúsculas

2.2.3.2 OperadorOrderBy

trewa.bd.sql.OperadorOrderBy

Esta clase define un operador válido para una expresión Order By. Los posibles operadores habilitados son:

OperadorWhere	Operador
ASCENDENTE	"ASC"

DESCENDENTE	"DESC"
-------------	--------

2.2.3.3 OperadorLogico

trewa.bd.sql.OperadorLogico

Esta clase define un operador lógico usado entre expresiones Where. Los posibles operadores habilitados son:

OperadorWhere	Operador
AND	"AND"
OR	"OR"

2.2.4 Expresiones

2.2.4.1 ExpresionWhere

trewa.bd.sql.ExpresionWhere

Clase que define una expresión simple en una cláusula where.

Constructores

→ `ExpresionWhere(Campo campoNuevo, OperadorWhere opNuevo, Object sValNuevo)`

Crea una expresión where con formato <campo> + <Operador> + <valor>.

→ `ExpresionWhere(Campo campoNuevo, OperadorWhere opNuevo)`

Crea una expresión where con formato <campo> + <Operador>.

Este constructor habilita la creación de Expresiones que incluyan operadores del tipo "IS NULL" , "IS NOT NULL".

→ `ExpresionWhere(Campo campoNuevo, ArrayList sValNuevo)`

Crea una expresión where con formato <campo> + OperadorWhere.OP_IN + <Lista de valores>.

Este constructor habilita la creación de expresiones que incluyan el operador "**IN**" acompañado de una lista de valores.

2.2.4.2 ExpresionOrderBy

trewa.bd.sql.ExpresionOrderBy

Clase que define una expresión en una cláusula Order By.

Constructores

→ `ExpresionOrderBy(Campo nuevoCampo, OperadorOrderBy nuevoOrden)`

Crea una expresión order by con formato <campo> + <Operador>

Ejemplo:

```
ExpresionOrderBy exp = new ExpresionOrderBy(TrMensaje.CAMPO_FECHA,OperadorOrderBy.ASCENDENTE );
```

2.2.5 Cláusulas

2.2.5.1 ClausulaWhere

trewa.bd.sql.ClausulaWhere

Clase que define una cláusula where para aplicar un filtro en algún método de la TrAPI. Una cláusula where puede incluir a otras cláusulas where y a expresiones where unidas por operadores lógicos ('and' u 'or').

Constructores

→ `ClausulaWhere()`

Crea una cláusula where sin expresiones pero con el operador lógico 'AND' por defecto como nexo de unión.

→ `ClausulaWhere(OperadorLogico nuevoOpLog)`

Crea una cláusula where sin expresiones que hace uso del operador lógico recibido como nexo de unión.

Métodos para inclusión de expresiones en la cláusula where

→ `boolean addExpresion(Campo campo, OperadorWhere op)`

Crea una expresión basada en operadores "IS NULL" o "IS NOT NULL" y la añade a la cláusula where.

→ `boolean addExpresion(Campo campo, OperadorWhere op, String strValor)`

Crea una expresión y la añade a la cláusula where.

→ `boolean addExpresionFecha(Campo campo, OperadorWhere op, TpoFecha fecha)`

Crea una expresión con valor de tipo fecha y la añade a la cláusula where.

→ `boolean addExpresionIn(Campo campo, ArrayList aValor)`

Crea una expresión basada en el operador "IN" y la añade a la cláusula where.

→ `boolean addExpresion(ExpresionWhere nuevaExpresion)`

Añade una nueva expresión a la cláusula where.

→ `boolean addExpresion(ClausulaWhere cWhere)`

Añade otra clausula where a la clausula where actual.

Con este método se habilitan mecanismos de anidamiento en los filtros.

→ `boolean addExpresion (CampoUtils campoUtils, int valor)`

Permite añadir utilidades a la consulta, como por ejemplo el MAX_ROW en los tipos que se permitan

→ `boolean addExpresion(CampoUtils campoUtils, String valor)`

Permite añadir utilidades a la consulta, como por ejemplo el MAX_ROW en los tipos que se permitan

→ `boolean addExpresionIn(Campo campo, List avalor)`

Crea una expresión basada en el operador IN y la añade a la cláusula where.

→ `boolean addExpresionIn(Campo campo, OperadorWhere opNuevo, ArrayList aValor)`

Crea una expresión basada en el operador IN y la añade a la cláusula where

→ `boolean addExpresionIn(Campo campo, OperadorWhere opNuevo, List aValor)`

Crea una expresión basada en el operador IN y la añade a la cláusula where

Ejemplos de uso de ClausulaWhere:

```
cWhere.addExpresion(TrMensaje.CAMPO_FECHA,OperadorWhere.OP_MAYOR_IGUAL,"10/10/2004");
```

```
cWhere.addExpresion(TrMensaje.CAMPO_REFMENSAJE, OperadorWhere.OP_IS_NOT_NULL);
```

2.2.5.2 ClausulaOrderBy

trewa.bd.sql.ClausulaOrderBy

Clase que define una cláusula order by para aplicar ordenación en algún método de la TrAPI.

Constructores

→ ClausulaOrderBy()

Crea una cláusula order by sin expresiones.

Métodos para inclusión de expresiones en la cláusula order by

→ boolean addExpresion(Campo campo, OperadorOrderBy op)

Crea una expresión order by y la añade a la cláusula order by.

Ejemplo:

```
// ORDER BY CAMPO_FECHA DESC, CAMPO_TEXTOMENSAJE ASC
ClausulaOrderBy orderMsj = new ClausulaOrderBy();
orderMsj.addExpresion(TrMensaje.CAMPO_FECHA,OperadorOrderBy.DESCENDENTE);
orderMsj.addExpresion(TrMensaje.CAMPO_TEXTOMENSAJE,OperadorOrderBy.ASCENDENTE);
```

→ boolean addExpresion(ExpresionOrderBy nuevaExpresion)

Añade una nueva expresión a la cláusula order by.

Ejemplo:

```
// ORDER BY CAMPO_FECHA DESC
ClausulaOrderBy orderMsj = new ClausulaOrderBy();
ExpresionOrderBy exp = new ExpresionOrderBy(TrMensaje.CAMPO_FECHA,OperadorOrderBy.DESCENDENTE );
orderMsj.addExpresion( exp );
```

2.2.5.3 ClausulaExclusion

trewa.bd.sql.ClausulaExclusion

Constructores

→ ClausulaExclusion()

Crea una nueva clausula de exclusiones.

Clase que define una cláusula de exclusión de entidades para en algún método de la TrAPI. Con ésta cláusula podemos definir qué datos queremos que nos devuelva una api, ya que al añadir exclusiones a una ClausulaExclusion se realizan consultas dinámica sólo con las entidades que no se hayan excluido con lo que se mejora el rendimiento pero no se obtienen los datos que excluyamos.



Métodos para inclusión de exclusiones

→ boolean addExclusion(CampoExclusion campoExc)

Añade una exclusión a la cláusula para indicar a la api que no queremos obtener datos de dicha entidad.

Ejemplo:

```
ClausulaExclusion exclusiones = new ClausulaExclusion ();  
exclusiones.addExclusion(TrExplorador.EXCL_CADUCIDADES);  
exclusiones.addExclusion(TrExplorador.EXCL_TIPOEXPEDIENTE);
```

2.2.6 Clases de Excepción.

Las TrAPIs avisan de errores producidos en las mismas, disparando excepciones del tipo TrException.

2.2.6.1 TrException

trewa.exception.TrException

Clase de excepción para avisar de errores producidos en las TrAPIs.

Métodos

→ String getMessage()

Obtiene el mensaje de error asociado a la excepción.

Ejemplo:

```
try{  
    ...  
} catch( TrException ex ){  
    System.out.println( ex.getMessage() );  
}
```

→ long getErrorCode()

Obtiene el código de error Trew@.

Ejemplo:

```
try{  
    ...  
} catch( TrException ex ){  
    System.out.println( "--> " + ex.getErrorCode () + " -- " + ex.getMessage() );  
}
```




2.2.6.2 TrExceptionMensajeCondicion

trewa.exception.TrExceptionMensajeCondicion extends *TrException*

Clase de excepción mediante la cual se pueden obtener los mensajes asociados a las condiciones y acciones cuando alguna de las apis que devuelven estos mensajes lance una excepción. Las apis que lanzan una instancia de *TrExceptionMensajeCondicion* en caso de que fallen porque no se cumple alguna condición obligatoria son las siguientes:

- deshacerTramitacion_aux
- generarDocumentoMultiple
- incorporarDocExternoMultiplesExpedientes
- incorporarDocumento
- incorporarDocumentoExterno
- incorporarDocumentoMultiplesExpedientes
- iniciarTareaExpediente
- recorrerDeshacerPaso
- tramitarExpedienteAux

Métodos

→ String getMessage()

Obtiene el mensaje de error asociado a la excepción.

→ long getErrorCode()

Obtiene el código de error Trew@.

→ TrMensajeCondicionAccion[] getMensajeCondicion()

Devuelve el array de mensajes de la condición/es que no se cumple.

Ejemplo:

```
try{
    apiUI.tramitarExpediente(...);
} catch( TrException ex ){
    if (ex instanceof TrExceptionMensajeCondicion) {
        //bucle para cada instancia
        ...
        LOG.error((ex(TrExceptionMensajeCondicion)).getMensajeCondicion[i]);
        ...
    }
    else {
        LOG.error(ex.getMessage());
    }
}
```

2.2.7 Calendarios hábiles

Para configurar que un sistema funcione con un calendario de días hábiles para realizar el cálculo de caducidades de plazo compuesto se debe dar de alta una constante en el sistema cuyo código debe ser COMP_CALEDAR.

El valor de ésta constante debe ser el nombre del componente que tiene la configuración del calendario.

El componente debe tener a su vez los siguientes datos de componente:

- **ACTIVADO:** Sus valores pueden ser TRUE o FALSE e indica si se usa el calendario de días hábiles par el cálculo de fechas de caducidad o si se hace con días naturales. Antes de que existiera esta posibilidad todos los cálculos de caducidades de plazos compuestos se realizaban con días naturales.
- **SABADO_HABIL:** Sus valores pueden ser TRUE o FALSE e indica si el sábado se considera día hábil para tenerlo en cuenta como festivo o no en el cálculo de plazos.

Por ejemplo en las siguientes capturas se puede ver en la configuración del sistema, la constante COMP_CALEDAR cuyo valor es CALEDAR (nombre del componente). En este caso el uso del calendario de días hábiles estaría activado y los sábados no se consideran hábiles.

Código	Denominación	Valor	Sistema
COMP_CALEDAR	Componente para calendarios hábiles	CALENDAR	TEST

La definición del componente y sus datos de componente

Nombre	Descripción	Dirección IP	Usuario	Password	Cód.w@ndA	Organismo donde está
CALENDAR	COMPONENTE PARA CALEDARIOS					CONSEJERÍA DE ... - (SEVILLA)

Atributo	Valor
ACTIVADO	TRUE
SABADO_HABIL	FALSE

2.2.8 Selector de plantillas múltiples

Para poder seleccionar automáticamente una de las platillas offices asociadas aun tipo de documento se deben crear selectores de platillas offices. Su funcionamiento es similar al de las condiciones, acciones y avisos pero el valor devuelto debe ser el código de la plantilla asociado al tipo de documento que se desea generar.

Selectores para plantillas office

Registros 1 a 3 de 3

Nombre	Descripción	Paquete	Función	Parámetros
CONS_DOC_GEN	CONSULTA_DOCUMENTO_GENERADO	es.juntadeandalucia.trewa.consultasPlantillas.consultaPlantillaMultiple	calculaPlantilla	(Exp, Documento Permitted, Exp Fase, Proc, Usuario, Tipo Documento)
CONS_MULT	CONSULTA MULTIPLANTILLA	trewa.test.ConsultaPlantillaTest	obtenerCodPlantilla	(Exp, Documento Permitted, Proc, Fecha, Usuario, Tipo Documento)

Para indicar que un tipo de documento tiene una consulta asociada, se hará mediante las dos nuevas propiedades de este elemento. Para su correcto funcionamiento de deberá seleccionar una consulta del listado disponible para el sistema y marcar el check como válida.

Selector Plantilla Office

¿Consulta válida?:

2.3 Datos de auditoría

TrAuditoria

trewa.bd.tpo.TrAuditoria

Clase que recoge los datos de auditoría. Se incluye en algunos tipos devueltos por TrAPIUI y TrAPIADM.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CREADO	String	Usuario que crea el registro
→ FECHACREADO	Timestamp	Fecha de creación del registro
→ MODIFICADO	String	Usuario que modifica el registro
→ FECHAMODIFICA	Timestamp	Fecha de modificación del registro.

2.4 JTrApis

2.4.1 Descripción de la TrAPIUI

2.4.1.1 Descripción de clases involucradas

2.4.1.1.1 Interfaz J-TrAPIUI

La interfaz J-TrAPIUI define todos los métodos de acceso al motor de tramitación TREW@. En la librería este acceso se realiza a través de la implementación de la interfaz **TrAPIUI** que se describe en este apartado.

2.4.1.1.1.1 TrAPIUI

trewa.bd.trapi.trapiui.TrAPIUI

Interfaz de acceso al motor de tramitación TREW@. Todos los métodos quedan descritos en el apartado "**Métodos del TrAPIUI**".

Para poder trabajar con la TrAPIUI es preciso crear una instancia de la misma a través de la clase Factoría: **TrAPIUIFactory**, descrita en el apartado siguiente.

2.4.1.1.2 Clase Factoría

La clase Factoría se encarga de crear una instancia correcta de la J-TrAPIUI de forma que los métodos de esta última aparezcan disponibles para las aplicaciones.

2.4.1.1.2.1 TrAPIUIFactory

trewa.bd.trapi.trapiui.TrAPIUIFactory

Clase que aporta funcionalidad para la correcta creación de una instancia de la TrAPIUI.

Métodos para la creación de instancias del TrAPIUI

→ TrAPIUI crearAPIUI(String stma)

Devuelve una instancia del TrAPIUI. La información para el establecimiento de la conexión la obtiene a través del perfil "default".

Entradas *String stma* *Nombre del sistema con el que va a trabajar el TrAPIUI.*
Si se pasa un valor *null* se creará un TrAPIUI sin sistema asociado y habrá métodos del TrAPIUI que no funcionarán correctamente. Se deben establecer los parámetros posteriormente a través del método *establecerConfiguracionSistema* del TrAPIUI.

Salidas *TrAPIUI* *Si el sistema pasado no existe devolverá null.*
Instancia del TrAPIUI o "null" si no se pudo crear.

→ TrAPIUI crearAPIUI(String usuario, String clave, String stma)

Devuelve una instancia del TrAPIU. La información para el establecimiento de la conexión la obtiene a través del perfil "default" pero en este caso hace uso del usuario y password que recibe como parámetros.

Entradas *String usuario* Usuario
String clave Clave de usuario
String stma *Nombre del sistema con el que va a trabajar el TrAPIUI.*
Si se pasa un valor *null* se creará un TrAPIUI sin sistema asociado y habrá métodos del TrAPIUI que no funcionarán correctamente. Se deben establecer los parámetros posteriormente a través del método *establecerConfiguracionSistema* del TrAPIUI.

Salidas *TrAPIUI* *Si el sistema pasado no existe devolverá null.*
Instancia del TrAPIUI o "null" si no se pudo crear.

→ TrAPIUI crearAPIUI(String strPerfil, String stma)

Devuelve una instancia del TrAPIU. La información para el establecimiento de la conexión la obtiene a través del perfil especificado en la llamada como parámetro.

Entradas *String strPerfil* *Nombre del perfil para el establecimiento de la conexión.*
String stma *Nombre del sistema con el que va a trabajar el TrAPIUI.*
Si se pasa un valor *null* se creará un TrAPIUI sin sistema asociado y habrá métodos del TrAPIUI que no funcionarán correctamente. Se deben establecer los parámetros posteriormente a través del método *establecerConfiguracionSistema* del TrAPIUI.

Salidas *TrAPIUI* *Si el sistema pasado no existe devolverá null.*
Instancia del TrAPIUI o "null" si no se pudo crear.

→ TrAPIUI crearAPIUI(String strPerfil, String usuario, String clave, String stma)

Devuelve una instancia del TrAPIU. La información para el establecimiento de la conexión la obtiene a través del perfil especificado en la llamada como parámetro, pero en este caso hace uso del usuario y password que recibe como parámetros.

Entradas *String strPerfil* *Nombre del perfil para el establecimiento de la conexión.*
String usuario Usuario
String clave Clave de usuario



String stma

Nombre del sistema con el que va a trabajar el TrAPIUI.

Si se pasa un valor *null* se creará un TrAPIUI sin sistema asociado y habrá métodos del TrAPIUI que no funcionarán correctamente. Se deben establecer los parámetros posteriormente a través del método *establecerConfiguracionSistema* del TrAPIUI. Si el sistema pasado no existe devolverá *null*.

Salida

TrAPIUI

Instancia del TrAPIUI o "null" si no se pudo crear.

→ TrAPIUI crearAPIUI(Properties properties, String stma)

Devuelve una instancia del TrAPIUI estableciendo una conexión por defecto. La información la obtiene a través del properties especificado en la llamada como parámetro.

Entradas

Properties

Properties con la configuración necesaria para la creación del TrAPIUI

properties

String stma

Nombre del sistema con el que va a trabajar el TrAPIUI.

Si se pasa un valor *null* se creará un TrAPIUI sin sistema asociado y habrá métodos del TrAPIUI que no funcionarán correctamente. Se deben establecer los parámetros posteriormente a través del método *establecerConfiguracionSistema* del TrAPIUI. Si el sistema pasado no existe devolverá *null*.
Instancia del TrAPIUI o "null" si no se pudo crear.

Salidas

TrAPIUI

→ TrAPIUI crearPooledAPIUI(String strPerfil, String stma)

Creamos el TrAPIUI con pool

Entradas

String strPerfil

Nombre del perfil para el establecimiento de la conexión.

String stma

Nombre del sistema con el que va a trabajar el TrAPIUI.

Si se pasa un valor *null* se creará un TrAPIUI sin sistema asociado y habrá métodos del TrAPIUI que no funcionarán correctamente. Se deben establecer los parámetros posteriormente a través del método *establecerConfiguracionSistema* del TrAPIUI. Si el sistema pasado no existe devolverá *null*.
Instancia del TrAPIUI o "null" si no se pudo crear.

Salidas

TrAPIUI

→ TrAPIUI crearPooledAPIUI(String strPerfil, String strUsr, String strPwd, String stma)

Creamos el TrAPIUI con pool

Entradas

String strPerfil

Nombre del perfil para el establecimiento de la conexión.

String strUsr

Usuario

String strPwd

Clave

String stma

Nombre del sistema con el que va a trabajar el TrAPIUI.

Si se pasa un valor *null* se creará un TrAPIUI sin sistema asociado y habrá métodos del TrAPIUI que no funcionarán correctamente. Se deben establecer los parámetros posteriormente a través del método *establecerConfiguracionSistema* del TrAPIUI.

Si el sistema pasado no existe devolverá *null*.

Salidas

TrAPIUI

Instancia del TrAPIUI o "null" si no se pudo crear.

Ejemplo de uso

En el siguiente ejemplo se muestra cómo obtener una instancia del TrAPIUI a través de la clase TrAPIUIFactory:

```
TrAPIUI apiUI = null;
TpoPK pK = null;

apiUI = TrAPIUIFactory.crearAPIUI( "perfilTrewa", "SISTEMA_TREWA" );
if ( apiUI != null ){
    // Ya se puede acceder a los métodos del TrAPIUI
    pK = new TpoPK( BigDecimal.valueOf(39) );
    try{
        TrParrafo[] arrParrafos = apiUI.obtenerParrafosDocumento( pK, null, null );
        ...
    }
}
catch(TrException ex)
{
    ex.printStackTrace();
}
}
```


2.4.1.2 Clases de entidad

Estas clases son el mecanismo de intercambio de información entre las aplicaciones y el motor Trew@ a través de las J-TrAPIs.

Las clases de entidad asociadas al TrAPIUI se encuentran en el paquete **trewa.bd.trapi.trapiui.tpo**.

2.4.1.2.1 TrAccion

trewa.bd.trapi.trapiui.tpo.TrAccion

Clase que representa la información asociada a una acción.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ NOMBRE	String	Nombre de la acción
→ DESCRIPCION	String	Descripción de la acción
→ STMA	TrSistema	Sistema

2.4.1.2.2 TrAccionDocumento

trewa.bd.trapi.trapiui.tpo.TrAccionDocumento

Clase que representa la información relacionada con una acción asociada a un documento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ ACCION	TrAccion	Acción
→ COMPROBAR	String	Comprobación → "G" – La acción se produce al Generar el documento → "I" – La acción se produce al Incorporar el documento → "T" – La acción se produce al Generar e Incorporar el documento

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la acción
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la acción
→ CAMPO_COMPROBAR	CADENA(1)	Comprobación → "G" – La acción se produce al Generar el documento → "I" – La acción se produce al Incorporar el documento → "T" – La acción se produce al Generar e Incorporar el documento
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema asociado a la acción

Métodos para aplicación de los filtros → obtenerAccionesDocumento

2.4.1.2.3 TrAccionDocumentoPortafirmas

trewa.bd.trapi.trapiui.tpo.TrAccionDocumentoPortafirmas

Clase que representa la información de una acción que se ejecutará al cambiar el estado del documento en Port@firmas. Esta información se envía a portamiras mediante el método enviaDocumentoExpediente y portamiras se encargará de ejecutar la acción cuando se cumpla el estado indicado.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ ESTADO	String	Estado del documento cuando se ejecutará la acción. → "FIRMADO" – Firmado

→ ACCION	String	→ "DEVUELTO" – Devuelto
→ TIPO	String	→ "LEIDO" – Leído
		→ "NUEVO" – Nuevo
		Procedimiento o url que se ejecutará.
		Tipo de la acción
		→ "PLSQL" – Procedimiento PLSQL
		→ "WEB" – Url web.

2.4.1.2.4 TrAccionTransicion

trewa.bd.trapi.trapiui.tpo.TrAccionTransicion

Clase que representa la información referente a acciones a realizar en una transición.

El modelo de referencia w@nda define una acción en transición como *"aquellas acciones que son realizadas por el tramitador de procedimientos sobre la base de unas condiciones establecidas, cuando se produce una transición concreta entre fases"*.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ ACCION	TrAccion	Acción
→ COMPROBAR	String	Comprobación
		→ "D" – La acción se produce al Deshacer la transición
		→ "T" – La acción se produce al Tramitar la transición
		→ "A" – La acción se produce al Tramitar y Deshacer la transición

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la acción
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la acción
→ CAMPO_COMPROBAR	CADENA(1)	Comprobación
		→ "D" – La acción se produce al Deshacer la transición
		→ "T" – La acción se produce al Tramitar la transición
		→ "A" – La acción se produce al Tramitar y Deshacer la transición
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema asociado a la acción

→ obtenerAccionesTransicion

Métodos para aplicación de los filtros

2.4.1.2.5 TrAvisoCaducidad

trewa.bd.trapi.trapiui.tpo.TrAvisoCaducidad

Clase que representa la información referente a un aviso establecido sobre una caducidad de un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CADUCIDAD	TrCaducidad	
→ FECHA	Timestamp	Fecha en la que se dio de alta el aviso
→ FECHA AVISO	Timestamp	Fecha de aviso de la caducidad al usuario
→ REFAVCAD	TpoPK	Referencia aviso caducidad
→ USUARIO	String	Usuario al que hay que avisar
→ TRANSICION	TrTransicion	Transición en la que hay que avisar al usuario
→ USUARIO AV	String	Usuario al que creo el aviso

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_FECHA	DATE	Fecha en la que se dio de alta el aviso
→ CAMPO_FECHA AVIS O	DATE	Fecha de aviso de la caducidad al usuario
→ CAMPO_USUARIO	CADENA(10)	Usuario al que hay que avisar
→ CAMPO_DESCTRANS	CADENA(50)	Descripción de la transición en la que hay que avisar al usuario
→ CAMPO_REFTIPOACT O	NUMERICO	Identificador del tipo de acto administrativo que representa la transición en la que hay que avisar al usuario.

→ obtenerAvisosCaducidadExpediente

→ obtenerAvisosCaducidadTransicion

Métodos para aplicación de los filtros

→ establecerAvisoCaducidadExpediente

2.4.1.2.6 TrBloque

trewa.bd.trapi.trapiui.tpo.TrBloque

Clase que representa la información referente a un bloque o módulo al que se puede acceder en cualquier momento del procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFBLOQUE	TpoPK	Identificador del bloque
→ NOMBRE	String	Nombre del bloque
→ DESCRIPCION	String	Descripción del bloque
→ TIPO	String	Tipo de bloque → "W" – Web → "R" – Informe → "F" – Pantalla
→ STMA	TrSistema	Sistema
→ INFORMAR	String	Indica si se informa al bus: → "S" – Sí → "N" – No
→ FORMULA	String	Código de Formul@
→ ACCIONINICIO	String	Acción de inicio de Formul@
→ ACCIONRETORNO	String	Acción de retorno de Formul@
→ PROCESA	String	Código de Proces@

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFBLOQUE	NUMERICO	Identificador del bloque
→ CAMPO_NOMBRE	CADENA(20)	Nombre del bloque
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del bloque
→ CAMPO_TIPO	CADENA(1)	Tipo del bloque → "W" – Web → "R" – Informe → "F" – Pantalla
→ CAMPO_REFSTMA	NUMERICO	Identificador del sistema
→ CAMPO_INFORMAR	CADENA(1)	Indica si se informa al bus: → "S" – Sí → "N" – No
→ CAMPO_FORMULA	CADENA(100)	Código de Formul@
→ CAMPO_ACCIONINICIO	CADENA(100)	Acción de inicio de Formul@

- **CAMPO_ACCIONRET ORNO** CADENA(100) Acción de retorno de Formul@
- **CAMPO_PROCESA** CADENA(100) Código de Proces@

→ obtenerBloquesDefinidos

Métodos para aplicación de los filtros

→ obtenerBloquesPermitidos

2.4.1.2.7 TrBloqueExplorador

trewa.bd.trapi.trapiui.tpo.TrBloqueExplorador

Clase que representa la información referente a las tareas pendientes de tipo “MANIPULAR_DATOS” definidas en una fase de un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ DESCRIPCION	String	Descripción del bloque permitido
→ OBLIGATORIO	String	Indica si es obligatoria la realización de esta tarea → “ S ” – Sí → “ N ” – No
→ REFBLOQUEPER	TpoPK	Identificador del bloque permitido.
→ ORDEN	Integer	Orden del bloque permitido.

2.4.1.2.8 TrBloquePermitido

trewa.bd.trapi.trapiui.tpo.TrBloquePermitido

Clase que representa la información referente a las tareas de tipo “MANIPULAR_DATOS” definidas en una fase de un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ BLOQUEFIN	TrBloque	Bloque final
→ ETIQUETA	String	Etiqueta del bloque permitido
→ DESCRIPCION	String	Descripción del bloque permitido
→ ETIQLARGA	String	Etiqueta larga para la tarea en fase
→ INFORMAR	String	Indica si se informa al bus de esta tarea → “ S ” – Sí → “ N ” – No
→ OBLIGATORIO	String	Indica si es obligatoria la realización de esta tarea → “ S ” – Sí → “ N ” – No
→ PLAZO	TrPlazo	Plazo de realización de la tarea en fase
→ REFBLOQUEPER	TpoPK	Identificador del bloque permitido.
→ BLOQUEINI	TrBloque	Bloque inicial
→ ORDEN	Integer	Orden del bloque permitido.
→ TIPOACTO	TrTipoActo	Tipo de acto.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFBLOQUE	NUMERICO	Identificador del bloque
→ CAMPO_ETIQUETA	CADENA(11)	Etiqueta del bloque permitido
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción del bloque permitido
→ CAMPO_TIPO	CADENA(1)	Tipo del bloque → “ W ” – Web → “ R ” – Informe → “ F ” – Pantalla
→ CAMPO_NOMBRE	CADENA(20)	Nombre del bloque
→ CAMPO_ETIQLARGA	CADENA(25)	Etiqueta larga para la tarea en fase
→ CAMPO_INFORMAR	CADENA(1)	Indica si se informa al bus de esta tarea → “ S ” – Sí → “ N ” – No
→ CAMPO_OBLIGATORIO	CADENA(1)	Indica si es obligatoria la realización de esta tarea → “ S ” – Sí → “ N ” – No

→	CAMPO_UNIDAD	CADENA(1)	Indica el tipo de unidad de tiempo → “ D ” – Días → “ M ” – Meses → “ A ” – Años
→	CAMPO_NUMUNIDADES	NUMERICO	Indica el número de unidades
→	CAMPO_DESCFECHALIMITE	CADENA(100)	Descripción del significado de la fecha límite de realización de la tarea.
→	CAMPO_REFBLOQUEINI	NUMERICO	Identificador del bloque inicial
→	CAMPO_ETIQUETABLQINI	CADENA(11)	Etiqueta del bloque inicial
→	CAMPO_DESCRIPCIONBLQINI	CADENA(50)	Descripción del bloque inicial
→	CAMPO_NOMBREBLQINI	CADENA(20)	Nombre del bloque inicial
→	CAMPO_TIPOBLQINI	CADENA(1)	Tipo del bloque → “ W ” – Web → “ R ” – Informe → “ F ” – Pantalla
→	CAMPO_ORDEN	NUMERICO	Orden del bloque permitido.
→	CAMPO_ABREVTIPOACTO	CADENA(11)	Abreviatura del tipo de acto.
→	CAMPO_DESCTIPOACTO	CADENA(50)	Descripción tipo de acto.
→	CAMPO_REFSTMATIPOACTO	NUMERICO	Referencia del sistema de tipo de acto.
→	CAMPO_REFTIPOACTO	NUMERICO	Referencia del tipo de acto. → obtenerBloquesPermitidos

Métodos para aplicación de los filtros

2.4.1.2.9 TrCaducidad

trewa.bd.trapi.trapiui.tpo.TrCaducidad

Clase que representa la información referente a caducidades que se pueden definir para un procedimiento.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	REFCADU	TpoPK	Identificador de la caducidad
→	ABREVIATURA	String	Abreviatura de la caducidad
→	DESCRIPCION	String	Descripción de la caducidad
→	UNIDAD	String	Unidad de medida de tiempo → “ D ” – Días → “ M ” – Meses → “ A ” – Años
→	NUMUNIDADES	long	Número de unidades que se cuentan (nº de días, nº de años,...)
→	TIPO	String	Tipo de caducidad → “ P ” - Plazo de ejecución → “ O ” - Otro
→	VIGENTE	String	Indica si la caducidad está vigente o activa en el procedimiento

- “S” – Sí
- “N” – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCADU	NUMERICO	Identificador de la caducidad
→ CAMPO_ABREVIATURA	CADENA(10)	Abreviatura de la caducidad
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción de la caducidad
→ CAMPO_UNIDAD	CADENA(1)	Unidad de medida de tiempo → “D” – Días → “M” – Meses → “A” – Años
→ CAMPO_NUMUNIDADES	NUMERICO(2)	Número de unidades que se cuentan (nº de días, nº de años,...)
→ CAMPO_TIPO	CADENA(1)	Tipo de caducidad → “P” - Plazo de ejecución → “O” - Otro
→ CAMPO_VIGENTE	CADENA(1)	Indica si la caducidad está vigente o activa en el procedimiento → “S” – Sí → “N” – No

- obtenerCaducidadesDefProcedimiento
- establecerAvisoCaducidadExpediente
- obtenerAvisosCaducidadTransicion
- obtenerCaducidadesExpediente

Métodos para aplicación de los filtros

2.4.1.2.10 TrCaducidadExpediente

trewa.bd.trapi.trapiui.tpo.TrCaducidadExpediente

Clase que representa la información referente de caducidades asociadas a un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFCADEXP	TpoPK	Identificador de la caducidad del expediente
→ CADUCIDAD	TrCaducidad	Caducidad que representa
→ FECHA	Timestamp	Fecha en la que comenzó a contar el tiempo para la caducidad
→ FECHALIMITE	Timestamp	Límite de tiempo para la caducidad
→ REFEXP	TpoPK	Identificador del expediente

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCADEXP	NUMÉRICO	Identificador de la caducidad del expediente
→ CAMPO_ABREVIATURA	CADENA(10)	Abreviatura de la caducidad
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción de la caducidad
→ CAMPO_TIPO	CADENA(1)	Tipo de caducidad → “P” - Plazo de ejecución → “O” - Otro
→ CAMPO_FECHA	DATE	Fecha en la que comenzó a contar el tiempo para la caducidad
→ CAMPO_FECHALIMITE	DATE	Límite de tiempo para la caducidad
→ CAMPO_REFCADU	NUMÉRICO	Identificador de la caducidad que representa (la definida en el procedimiento)
→ CAMPO_REFEXP	NUMÉRICO	Identificador del expediente

Métodos para aplicación de los filtros

- obtenerCaducidadesExpediente
- obtenerExpedientesCaducados

2.4.1.2.11 TrCambioProcedimientoExpediente

trewa.bd.trapi.trapiui.tpo.TrCambioProcedimientoExpediente

Clase que representa la información referente a los datos de un procedimiento que ha seguido o sigue un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOEXP	TpoPK	Identificador del tipo de expediente
→ REFDEFPROC	TpoPK	Identificador de la definición de procedimiento que sigue o ha seguido el expediente
→ OBSERVACIONES	String	Observaciones anotadas en la “historia de evoluciones” por las que ha pasado el expediente
→ USUARIO	String	Usuario que dio de alta el expediente o cambió el procedimiento y tipo de expediente del expediente
→ FECHA	Timestamp	Fecha de alta o cambio de flujo del expediente en TREW@
→ VIGENTE	String	Indica si los datos están vigentes o no → “S” – Los datos sí están vigentes → “N” – Los datos no están vigentes
→ NOMBREUSU	String	Nombre y apellidos del usuario

- **DESCTIPOEXP** String Descripción del tipo de expediente
- **DESCDEFPROC** String Descripción de la definición del procedimiento

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOEXP	NUMÉRICO	Identificador del tipo de expediente
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición de procedimiento que sigue o ha seguido el expediente
→ CAMPO_OBSERVACIONES	CADENA(250)	Observaciones anotadas en la "historia de evoluciones" por las que ha pasado el expediente
→ CAMPO_USUARIO	CADENA(10)	Usuario que dio de alta el expediente o cambió el procedimiento y tipo de expediente del expediente
→ CAMPO_FECHA	DATE	Fecha de alta o cambio de flujo del expediente en TREW@
→ CAMPO_VIGENTE	CADENA(1)	Indica si los datos están vigentes o no → "S" – Los datos sí están vigentes → "N" – Los datos no están vigentes
→ CAMPO_NOMBREUSUARIO	CADENA(98)	Nombre y apellidos del usuario
→ CAMPO_DESCTIPOEXP	CADENA(50)	Descripción del tipo de expediente
→ CAMPO_DESCDEFPROC	CADENA(50)	Descripción de la definición del procedimiento

Métodos para aplicación de los filtros

→ obtenerDatosExpediente

2.4.1.2.12 TrComponente

trewa.bd.trapi.trapiui.tpo.TrComponente

Clase que representa la información referente a componentes que pueden intervenir en w@nda.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFCOMPONENTE	TpoPK	Identificador del componente.
→ REFORGANISMO	TpoPK	Identificador del organismo en el que está el componente
→ NOMBRE	String	Nombre del componente
→ DESCRIPCION	String	Descripción del componente
→ TIPOCOMPONENTE	TrTipoComponente	Tipo de componente
→ CODWANDA	Long	Identificador del componente en w@nda
→ DIRECCIONIP	String	Dirección IP del componente.

- **USUARIO** String Usuario para acceder al componente.
- **PASSWORD** String Password del usuario para acceder al componente.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCOMPONENTE	NUMÉRICO	Identificador del componente
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo en el que está el componente
→ CAMPO_NOMBRE	CADENA(50)	Nombre del componente
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del componente
→ CAMPO_REFTIPOCOMPONENTE	NUMERICO	Identificador del tipo de componente
→ CAMPO_CODWANDA	NUMERICO	Identificador del componente en w@ndA
→ CAMPO_DIRECCIONIP	CADENA(255)	Dirección IP del componente.
→ CAMPO_USUARIO	CADENA(30)	Usuario para acceder al componente.
→ CAMPO_PASSWORD	CADENA(30)	Password del usuario para acceder al componente.

Métodos para aplicación de los filtros → obtenerComponentes

2.4.1.2.13 TrComunicacion

trewa.bd.trapi.trapiui.tpo.TrComunicacion

Clase que representa la información referente a una comunicación existente en el sistema.

Una comunicación podrá ser asociada a un tipo de evolución o a un interesado expediente concreto.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFCOMU	TpoPK	Identificador de la comunicación.
→ USUARIOENV	String	Usuario del sistema, normalmente tramitador, que realiza el envío de la comunicación al interesado expediente.
→ TITULO	String	Título de la comunicación enviada al interesado expediente.
→ CUERPO	String	Contenido del mensaje de la comunicación que el usuario envía al interesado expediente.
→ PRIORIDAD	String	Establece el nivel de prioridad que se asigna a la comunicación. Sus posibles valores son: → “ A ” – Alta → “ M ” – Media → “ B ” – Baja

→ FECHAINICIO	Timestamp	Fecha a partir de la cual la comunicación está vigente en el sistema.
→ FECHAFIN	Timestamp	Fecha hasta la cual la comunicación tiene vigencia en el sistema. Si no tiene ningún valor asignado se considera que la comunicación no tiene fin de vigencia.
→ LEIDA	String	Indica si la comunicación ha sido leída o no. → “ S ” – Sí. → “ N ” – No. Solo es aplicable para comunicaciones a interesados expediente.
→ FECHALECTURA	Timestamp	Fecha de lectura de la comunicación. Solo es aplicable para comunicaciones a interesados expediente.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCOMU	NUMÉRICO	Identificador de la comunicación.
→ CAMPO_USUARIOENV	CADENA(10)	Usuario que envía la comunicación.
→ CAMPO_TITULO	CADENA(50)	Título que tiene la comunicación.
→ CAMPO_CUERPO	CADENA(500)	Mensaje de la comunicación.
→ CAMPO_PRIORIDAD	CADENA(1)	Prioridad de la comunicación.
→ CAMPO_FECHAINICIO	DATE	Fecha de inicio de vigencia de la comunicación.
→ CAMPO_FECHAFIN	DATE	Fecha de fin de vigencia de la comunicación.
→ CAMPO_LEIDA	CADENA(1)	Indica si la comunicación ha sido leída o no por el interesado expediente al cual va dirigida.
→ CAMPO_FECHALECTURA	DATE	Fecha de lectura de la comunicación.

Métodos para aplicación de los filtros

- crearComunicacionGeneral
- crearComunicacionPersonal
- obtenerComunicacionesIntExp
- obtenerComunicacionesProcedimiento

2.4.1.2.14 TrCondicion

trewa.bd.trapi.trapiui.tpo.TrCondicion

Clase que representa la información referente a una condición.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
-----------------	-------------	--------------------

→ NOMBRE	String	Nombre de la condición
→ DESCRIPCION	String	Descripción de la condición
→ STMA	TrSistema	Sistema
		→ obtenerCondicionTransicion
Métodos para aplicación de los filtros		→ obtenerCondicionesDocumento

2.4.1.2.15 TrCondicionAccionAviso

trewa.bd.trapi.trapiui.tpo.TrCondicionAccionAviso

Clase que representa la información de una condición, acción o aviso.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ NOMBRE	String	Nombre de la condición, acción o aviso.
→ DESCRIPCION	String	Descripción de la condición, acción o aviso.
→ COMPROBAR	String	Indica cuando se comprueba la condición, acción o aviso.
→ OBLIGATORIA	String	Indica si es obligatoria. → “ S ” – Sí → “ N ” – No
→ TIPO	String	Tipo → “ C ” – Condición → “ A ” – Acción → “ W ” – Aviso
→ REFSTMA	TpoPK	Identificador del sistema
→ MENSAJEAVISO	String	Mensaje del aviso.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la condición, acción o aviso.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la condición, acción o aviso.
→ CAMPO_COMPROBAR	CADENA(1)	Indica cuando se comprueba la condición, acción o aviso.
→ CAMPO_OBLIGATORIA	CADENA(1)	Indica si es obligatoria. → “ S ” – Sí → “ N ” – No
→ CAMPO_TIPO	CADENA(1)	Tipo → “ C ” – Condición → “ A ” – Acción

- **CAMPO_REFSTMA** NUMERICO → **“W”** – Aviso
Identificador del sistema
- **CAMPO_MENSAJEAV ISO** CADENA(200) Mensaje del aviso

Métodos para aplicación de los filtros

→ obtenerCondicionesAccionesAvisos

2.4.1.2.16 TrCondicionDocumento

trewa.bd.trapi.trapiui.tpo.TrCondicionDocumento

Clase que representa la información referente a condiciones asociadas a un documento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CONDICION	TrCondicion	Condición
→ COMPROBAR	String	Comprobación → “G” – La condición se produce al Generar el documento → “I” – La condición se produce al Incorporar el documento → “T” – La condición se produce al Generar e Incorporar el documento → “V” – La condición se produce al Visualizar el documento
→ OBLIGATORIA	String	Indica si la condición es o no obligatoria → “S” – La condición es obligatoria → “N” – La condición no es obligatoria

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la condición
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la condición
→ CAMPO_COMPROBAR	CADENA(1)	Comprobación → “G” – La condición se produce al Generar el documento → “I” – La condición se produce al Incorporar el documento → “T” – La condición se produce al Generar e Incorporar el documento → “V” – La condición se produce al Visualizar el documento
→ CAMPO_OBLIGATORIA	CADENA(1)	Indica si la condición es o no obligatoria → “S” – La condición es obligatoria → “N” – La condición no es obligatoria
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema asociado a la condición

Métodos para aplicación de los filtros

→ obtenerCondicionesDocumento

2.4.1.2.17 TrCondicionTransicion

trewa.bd.trapi.trapiui.tpo.TrCondicionTransicion

Clase que representa la información referente a condiciones a evaluar en una transición.

El modelo de referencia w@nda define una condición de transición como “el conjunto de circunstancias que se evalúan por el sistema cada vez que se intenta producir una transición en el procedimiento”.

Atributos accesibles mediante métodos get/set

Atributo	Tipo	Descripción
→ CONDICION	TrCondicion	Condición
→ COMPROBAR	String	Comprobación → “ D ” – La condición se produce al Deshacer la transición → “ T ” – La condición se produce al Tramitar la transición → “ A ” – La condición se produce al Tramitar y Deshacer la transición → “ V ” – La condición se produce al Visualizar la transición
→ OBLIGATORIA	String	Indica si la condición es o no obligatoria → “ S ” – La condición es obligatoria → “ N ” – La condición no es obligatoria
→ MENSAJEOK	String	Mensaje a mostrar cuando se cumple la condición
→ MENSAJENOOK	String	Mensaje a mostrar cuando no se cumple la condición
→ MOSTRARM SJ	String	Mensaje a mostrar. Sus valores pueden ser: → “ O ” – Mensaje OK → “ N ” – La condición se produce al Visualizar la transición → “ A ” – La condición se produce al Tramitar y Deshacer la transición → “ I ” – La condición se produce al Visualizar la transición

Campos definidos para filtrados y ordenación

Campo	Tipo	Descripción y restricciones de uso
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la condición
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la condición
→ CAMPO_COMPROBAR	CADENA(1)	Comprobación → “ D ” – La condición se produce al Deshacer la transición → “ T ” – La condición se produce al Tramitar la transición → “ A ” – La condición se produce al Tramitar y Deshacer la transición → “ V ” – La condición se produce al Visualizar la transición (ver transiciones_posibles-tr_pr_transiciones_posibles)
→ CAMPO_OBLIGATORIO	CADENA(1)	Indica si la condición es o no obligatoria

A

- **CAMPO_REFSTMA** NUMÉRICO
- **CAMPO_MOSTRARMS** CADENA(1)

J

- **CAMPO_MENSAJEOK** CADENA(200)
- **CAMPO_MENSAJENO** CADENA(200)

OK

→ “**S**” – La condición es obligatoria

→ “**N**” – La condición no es obligatoria

Identificador del sistema asociado a la condición

Identificador del sistema asociado a la condición

Identificador del sistema asociado a la condición

Identificador del sistema asociado a la condición

Métodos para aplicación de los filtros

→ obtenerCondicionesTransicion

2.4.1.2.18 TrConstante

trewa.bd.trapi.trapiui.tpo.TrConstante

Clase que recoge la información de las constantes del motor de tramitación.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODCONSTANTE	String	Condición
→ DESCRIPCION	String	Descripción de constante
→ VALOR	String	Valor de constante
→ STMA	TrSistema	Referencia del sistema al que pertenece la constante

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODIGOCTE	CADENA(50)	Código de la constante
→ CAMPO_DESCRIPCIO	CADENA(250)	Descripción de la constantes
N		
→ CAMPO_REFSTMA	NUMÉRICO	Referencia del sistema de la constante.
→ CAMPO_VALOR	CADENA(50)	Valor de la contante

Métodos para aplicación de los filtros

→ obtenerConstante

2.4.1.2.19 TrConstanteGn

trewa.bd.trapi.trapiui.tpo.TrConstanteGn

Clase que recoge la información de las constantes generales.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	CODCONSTANTE	String	Condición
→	DESCRIPCION	String	Descripción de constante
→	VALOR	String	Valor de constante

Campos definidos para filtrados y ordenación

	<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_CODCONSTANTE	CADENA(10)	Código de la constante
→	CAMPO_DESCRIPCION	CADENA(64)	Descripción de la constantes
→	CAMPO_VALOR	CADENA(80)	Valor de la contante

Métodos para aplicación de los filtros → obtenerConstante

2.4.1.2.20 TrConsultaCertificado

trewa.bd.trapi.trapiui.tpo.TrConsultaCertificado

Clase que representa los datos de una consulta de un certificado de un interesado en un documento de un expediente.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	REFCONSULTA	TpoPK	Identificador de la consulta
→	REFDOCEXP	TpoPK	Identificador del documento del expediente
→	REFEXPEDIENTE	TpoPK	Identificador del expediente
→	REFINTERESADO	TpoPK	Identificador del interesado
→	REFRAZONINTEXP	TpoPK	Identificador de la razón de interés del interesado en el expediente
→	REFRAZONINTDOC	TpoPK	Identificador de la razón de interés del interesado en el documento
→	CODUSUARIO	String	Código del usuario
→	FECHA	Timestamp	Fecha de la consulta
→	CODIGO	String	Código de la consulta
→	DESCRIPCION	String	Descripción de la consulta
→	DATOS	byte[]	Datos de la consulta (típicamente XML)

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCONSULTA	NUMERICO	Identificador de la consulta
→ CAMPO_REFDOCEXP	NUMERICO	Identificador del documento del expediente
→ CAMPO_REFEXPEDIENTE	NUMERICO	Identificador del expediente
→ CAMPO_REFINTERESADO	NUMERICO	Identificador del interesado
→ CAMPO_REFRAZONINTEXP	NUMERICO	Identificador de la razón de interés del interesado en el expediente
→ CAMPO_REFRAZONINTDOC	NUMERICO	Identificador de la razón de interés del interesado en el documento
→ CAMPO_CODUSUARIO	CADENA(10)	Código del usuario
→ CAMPO_FECHA	DATE	Fecha de la consulta
→ CAMPO_CODIGO	CADENA(40)	Código de la consulta
→ CAMPO_DESCRIPCION	CADENA(2000)	Descripción de la consulta
→ CAMPO_DATOS	byte[]	Indica si la consulta del certificado tiene asociado datos binarios (típicamente XML). Si el valor de dicho campo es distinto de null se devolverá "S", si no "N". Por tanto, los valores que se pueden usar para aplicar este filtro son: - "S" - Para obtener los que tienen datos - "N" - Para obtener los que no tienen datos

Métodos para aplicación de los filtros

- obtenerConsultaCertificado
- obtenerConsultaCertificadoOracle

2.4.1.2.21 TrDatoComponente

trewa.bd.trapi.trapiui.tpo.TrDatoComponente

Clase que representa otros datos necesarios para el componente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDATOCOMP	TpoPK	Identificador del dato del componente
→ COMPONENTE	TrComponente	Componente
→ ATRIBUTO	String	Nombre del atributo del componente
→ VALOR	String	Valor del atributo del componente

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDATOCO MP	NUMERICO	Identificador del dato del componente
→ CAMPO_REFCOMP	NUMERICO	Identificador del componente
→ CAMPO_ATRIBUTO	CADENA(32)	Nombre del atributo del componente
→ CAMPO_VALOR	CADENA(255)	Valor del atributo del componente

Métodos para aplicación de los filtros

→ obtenerDatosComponente

2.4.1.2.22 TrDatosContacto

trewa.bd.trapi.trapiui.tpo.TrDatosContacto

Clase que representa los datos de contacto de los interesados.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDATOCONT	TpoPK	Identificador del dato de contacto
→ MUNICIPIO	TrMunicipio	Municipio
→ PAIS	TrPais	País
→ TIPOVIA	TrTipoVia	Tipo de vía
→ NOMBREVIA	String	Nombre de la vía
→ NUMERO	Integer	Número del domicilio
→ LETRA	String	Letra del domicilio
→ ESCALERA	String	Escalera del domicilio
→ PISO	Integer	Piso del domicilio
→ PUERTA	String	Puerta del domicilio
→ CODPOSTAL	Integer	Código postal del domicilio
→ TELEFONO	String	Número(s) de teléfono de contacto
→ TLFMOVIL	String	Número(s) de teléfono móvil
→ FAX	String	Número(s) de fax
→ EMAIL	String	Correo electrónico
→ PORDEFECTO	String	Indica si tiene un interesado por defecto → " S " – Tiene interesado por defecto → " N " – No tiene interesado por defecto
→ REGION	String	Región
→ CIUDAD	String	Ciudad

Campos definidos para filtrados y ordenación



<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDATOCOUNT	NUMERICO	Identificador del dato de contacto
→ CAMPO_CODMUNICIPIO	CADENA(3)	Código del municipio
→ CAMPO_MUNICIPIO	CADENA(32)	Nombre del municipio
→ CAMPO_CODPROVINCIA	CADENA(3)	Código de la provincia
→ CAMPO_PROVINCIA	CADENA(32)	Nombre de la provincia
→ CAMPO_CODPAIS	CADENA(3)	Código del país
→ CAMPO_PAIS	CADENA(32)	Nombre del país
→ CAMPO_TIPOVIA	CADENA(10)	Tipo de vía
→ CAMPO_NOMBREVIA	CADENA(100)	Nombre de la vía
→ CAMPO_NUMERO	NUMERICO	Número del domicilio
→ CAMPO_LETRA	CADENA(2)	Letra del domicilio
→ CAMPO_ESCALERA	CADENA(2)	Escalera del domicilio
→ CAMPO_PISO	NUMERICO	Piso del domicilio
→ CAMPO_PUERTA	CADENA(2)	Puerta del domicilio
→ CAMPO_CODPOSTAL	NUMERICO	Código postal del domicilio
→ CAMPO_TELEFONO	CADENA(25)	Número(s) de teléfono de contacto
→ CAMPO_TLFMOVIL	CADENA(25)	Número(s) de teléfono móvil
→ CAMPO_FAX	CADENA(25)	Número(s) de Fax
→ CAMPO_EMAIL	CADENA(64)	Correo electrónico
→ CAMPO_PORDEFECTO	CADENA(1)	Indica si tiene un interesado por defecto → "S" – Tiene interesado por defecto → "N" – No tiene interesado por defecto
→ CAMPO_REGION	CADENA(100)	Región
→ CAMPO_CIUADAD	CADENA(100)	Ciudad

- altaExpediente
- getNewTrDatosContacto
- insertarInteresadoExpediente
- modificarInteresado
- obtenerDatosContactoInteresado
- obtenerDatosContacto
- obtenerEmpleado
- obtenerOrganismo
- obtenerOrganismosExpediente
- obtenerOrganismosHijos
- obtenerOrganismosProcedimiento
- obtenerOrganismoTramitaExpediente
- obtenerPtoTrabOrganismo

Métodos para aplicación de los filtros



Junta de Andalucía

Agencia Digital de Andalucía

Trew@

Guía de referencia de JTrApi

2.4.1.2.23 TrDatosRegistroDocumentos

trewa.bd.trapi.trapiui.tpo.TrDatosRegistroDocumentos

Clase que representa los datos de registro de los documentos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODIGOREGISTRO	TpoPK	Identificador del registro
→ FECHAREGISTRO	Timestamp	Fecha del registro
→ REMITENTE	String	Remitente del registro
→ DESTINATARIO	String	Destinatario del registro
→ OFICINAENTRADA	String	Oficina de entrada del registro
→ OFICINASALIDA	String	Oficina de salida del registro
→ ASUNTO	String	Asunto del registro
→ TIPO	String	Indica si es de entrada o salida → “S” – Es de salida → “E” – Es de entrada
→ REGISTRO	String	Datos del registro (típicamente XML)
→ CONSEJERIA	String	Datos de la consejería
→ ORGANISMO	String	Datos del organismo
→ PROVINCIA	String	Datos de la provincia

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODREG	CADENA(20)	Identificador del dato de contacto
→ CAMPO_FECHAREGISTRO	DATE	Código del municipio
→ CAMPO_REMITENTE	CADENA(80)	Nombre del municipio
→ CAMPO_DESTINATARIO	CADENA(80)	Código de la provincia
→ CAMPO_OFICINASAL	CADENA(50)	Nombre de la provincia
→ CAMPO_OFICINAENT	CADENA(50)	Código del país
→ CAMPO_ASUNTO	CADENA(250)	Nombre del país
→ CAMPO_TIPOREG	CADENA(1)	Indica si es de entrada o salida → “S” – Es de salida → “E” – Es de entrada
→ CAMPO_CONSEJERIA	CADENA(50)	Nombre de la provincia
→ CAMPO_ORGANISMO	CADENA(50)	Nombre de la provincia
→ CAMPO_PROVINCIA	CADENA(50)	Nombre de la provincia

Métodos para aplicación de los filtros

→ obtenerRegistrosDocumento

2.4.1.2.24 TrDatoTipoPago

trewa.bd.trapi.trapiui.tpo.TrTipoPago

Clase que representa otros datos necesarios para los tipos de pago.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDATO_TIPOPAGO	TpoPK	Identificador del dato del tipo de pago
→ TIPOPAGO	TrTipoPago	Tipo de pago
→ ATRIBUTO	String	Nombre del atributo del tipo de pago
→ VALOR	String	Valor del atributo del tipo de pago

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDATO_TIPO PAGO	NUMERICO	Identificador del dato del tipo de pago
→ CAMPO_REFTIPOPAGO	NUMERICO	Identificador del tipo de pago
→ CAMPO_ATRIBUTO	CADENA(32)	Nombre del atributo del componente
→ CAMPO_VALOR	CADENA(255)	Valor del atributo del componente

Métodos para aplicación de los filtros

→ obtenerDatoTipoPago

2.4.1.2.25 TrDiaFestivo

trewa.bd.trapi.trapiui.tpo.TrDiaFestivo

Clase que representa un día festivo.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ fecha	Timestamp	Fecha del festivo
→ descripcion	String	Descripción
→ codProvincia	String	Código de la provincia
→ provincia	String	Descripción de la provincia
→ codMunicipio	Integer	Código del municipio
→ municipio	String	Descripción del municipio

→ anyo	Integer	Año
→ dia	Integer	Día
→ mes	Integer	Mes

Métodos con uso de la clase

→ obtenerDiasFestivos

2.4.1.2.26 TrDefProcedimiento

trewa.bd.trapi.trapiui.tpo.TrDefProcedimiento

Clase que representa la información referente a una definición de procedimientos en TREW@.

El modelo de referencia W@nda define la definición de procedimiento como “el resultado del modelado de un determinado procedimiento”.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoría
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento
→ ABREVIATURA	String	Abreviatura de la definición del procedimiento
→ DESCRIPCION	String	Descripción de la definición del procedimiento
→ STMA	TrSistema	Sistema
→ INFORMAR	String	Indica si se tendrá que informar al bus de este procedimiento y de los expedientes que se tramiten → “ S ” – Sí → “ N ” – No
→ VIGENTE	String	Indica si el procedimiento está vigente. → “ S ” – Si está vigente → “ N ” – No está vigente
→ CATEGORIA	String	Indica la categoría del procedimiento → “ F ” – Familia → “ S ” – Subfamilia → “ P ” – Procedimiento
→ DESCRIPCIONAMP	String	Descripción ampliada del procedimiento.
→ OTROSDATOS	String	Indica si el procedimiento tiene otros datos. → “ S ” – Sí → “ N ” – No
→ FAMILIA	TrFamiliaSubfamilia	Familia
→ SUBFAMILIA	TrFamiliaSubfamilia	Subfamilia
→ ORGANISMO	TrOrganismo	Organismo a la que pertenece.
→ ORGCOMPETENTE	TrOrganismo	Organismo competente.
→ ORGRESUELVE	TrOrganismo	Organismo que resuelve.
→ ORGTRAMITA	TrOrganismo	Organismo que tramita.

→	CODWANDA	String	Valor en w@ndA del procedimiento
→	COMENTARIOS	String	Comentarios al procedimiento/familia/subfamilia
→	USUARIOBLQ	String	Usuario que tiene bloqueado el procedimiento/familia/subfamilia
→	FINALIDAD	String	Finalidad de la definición de procedimiento, para soporte SCSP.
→	CODIGORPA	String	Identificador del procedimiento en el Registro de Procedimientos Administrativos
→	SERIEDOCARCH	String	Identificador de la serie documental para el archivo del expediente
→	CLASIFICACIÓN	String	Identificador del procedimiento administrativo al que pertenece el expediente.
→	SERVICIONOTI	String	Código de servicio Notific@ asociado a nivel de procedimiento.
→	QAA	String	Nivel de seguridad requerido para que los expedientes del procedimiento sean consultados.
→	NIVELINTCC	String	Nivel de integración con Carpeta Ciudadana del procedimiento.
→	ADAPTADOENI	String	Nivel de integración con Carpeta Ciudadana del procedimiento.

Campos definidos para filtrados y ordenación

	<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_REFDEFPROC	NUMERICO	Identificador de la definición del procedimiento.
→	CAMPO_ABREVIATUR A	CADENA(10)	Abreviatura de la definición del procedimiento
→	CAMPO_DESCRIPCION	CADENA(50)	Descripción de la definición del procedimiento
→	CAMPO_REFSTMA	NUMERICO	Identificador del sistema
→	CAMPO_SISTEMA	CADENA(10)	Nombre del sistema
→	CAMPO_INFORMAR	CADENA(1)	Indica si se tendrá que informar al bus de este procedimiento y de los expedientes que se tramiten → "S" – Sí → "N" – No
→	CAMPO_VIGENTE	CADENA(1)	Indica si el procedimiento está vigente. → "S" – Si está vigente → "N" – No está vigente
→	CAMPO_CATEGORIA	CADENA(1)	Indica la categoría del procedimiento → "F" – Familia → "S" – Subfamilia → "P" – Procedimiento
→	CAMPO_DESCRIPCION AMP	CADENA(250)	Descripción ampliada del procedimiento.
→	CAMPO_OTROS DATOS	CADENA(1)	Indica si el procedimiento tiene otros datos. → "S" – Sí → "N" – No
→	CAMPO_REFSUBFAMILIA	NUMERICO	Identificador de la familia/subfamilia
→	CAMPO_REFORGANISMO	NUMERICO	Identificador del organismo al que pertenece.
→	CAMPO_CODIGOORG	CADENA(16)	Código del organismo al que pertenece.
→	CAMPO_NOMBREORG	CADENA(32)	Nombre del organismo al que pertenece.
→	CAMPO_DESCORG	CADENA(200)	Descripción del organismo al que pertenece.



→	CAMPO_REFORGCAMP	NUMERICO	Identificador del organismo competente.
→	CAMPO_CODIGOORGCOMP	CADENA(16)	Código del organismo competente.
→	CAMPO_NOMBREORGCOMP	CADENA(32)	Nombre del organismo competente.
→	CAMPO_DESCORGCAMP	CADENA(200)	Descripción del organismo competente.
→	CAMPO_REFORGRES	NUMERICO	Identificador del organismo que resuelve.
→	CAMPO_CODIGOORGRES	CADENA(16)	Código del organismo que resuelve.
→	CAMPO_NOMBREORGRES	CADENA(32)	Nombre del organismo que resuelve.
→	CAMPO_DESCORGRES	CADENA(200)	Descripción del organismo que resuelve.
→	CAMPO_REFORGTRAM	NUMERICO	Identificador del organismo que tramita.
→	CAMPO_CODIGOORGRAM	CADENA(16)	Código del organismo que tramita.
→	CAMPO_NOMBREORGRAM	CADENA(32)	Nombre del organismo que tramita.
→	CAMPO_DESCORGRAM	CADENA(200)	Descripción del organismo que tramita.
→	CAMPO_CODWANDA	CADENA(15)	Valor en w@ndA del procedimiento
→	CAMPO_COMENTARIOS	CADENA(250)	Comentarios al procedimiento/familia/subfamilia
→	CAMPO_USUARIOBLQ	CADENA(10)	Usuario que tiene bloqueado el procedimiento/familia/subfamilia
→	CAMPO_CREADO	CADENA	Usuario que crea el registro
→	CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→	CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→	CAMPO_FECHAMODIFICA	DATE	Fecha de modificación del registro
→	CAMPO_CODIGORPA	CADENA(20)	Identificador del procedimiento en el Registro de Procedimientos Administrativos
→	CAMPO_SERIEDOC	CADENA(50)	Identificador de la serie documental para archivar el expediente
→	CAMPO_CLASIFICACION	CADENA(50)	Indica la clasificación del procedimiento o código SIA de procedimiento
→	CAMPO_SERVICIONOTI	CADENA(20)	Indica el código de servicio asociado en el Sistema de Notificaciones
→	INIT_ROW	NUMERICO	Indica el primer registro a devolver
→	MAX_ROW	NUMERICO	Indica el número máximo de filas a devolver
→	CAMPO_QAA	NUMERICO	Indica el nivel de seguridad de acceso a los expedientes del procedimiento.
→	CAMPO_NIVELINTCC	NUMERICO	Indica el nivel de integración con Carpeta Ciudadana del procedimiento.
→	CAMPO_QAA	NUMERICO	Indica si el procedimiento se encuentra adaptado para el eni. Sus valores pueden ser: → "S" – Se encuentra adaptado al eni → "N" – No se encuentra adaptado al eni



Métodos para aplicación de los filtros

- explorarExpedientesSimple
- generaExpedientesExplorador
- obtenerDatosFase
- obtenerDefProcedimientosDefinidos
- obtenerExpedientes
- obtenerExpedientesCaducados
- obtenerExpedientesSimple
- obtenerExpsRelacionInteresado
- obtenerFases
- obtenerFasesFinTransicion
- obtenerFasesProcedimiento
- obtenerInteresadoExpediente
- obtenerInteresadoExpedienteCC
- obtenerInteresadoExpedienteDadoDeBaja
- obtenerProcedimientosUsuario
- obtenerRelacionesDefinidas
- obtenerTipoExpAndTipoEvVersionDefProc
- obtenerVersionDefProcedimiento
- recorreFasesPosTipEv

2.4.1.2.27 TrDocMultiple

trewa.bd.trapi.trapiui.tpo.TrDocMultiple

Clase que recoge dos identificadores, el REFDOCEXP con le identificador del documento del expediente y el REFAUX con un identificador auxiliar a usar según convenga.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ FORMATO	String	Formato del documento
→ MODOGEN	String	Modo de generación del documento
→ NOMBREFICHERO	String	Nombre del fichero
→ REFAUX	TpoPK	Referencia de documento auxiliar
→ REFDOCEXP	TpoPK	Referencia del documento expediente

Métodos con uso de la clase

→ generarDocumento

2.4.1.2.28 TrDocumentoExpediente

trewa.bd.trapi.trapiui.tpo.TrDocumentoExpediente

Clase que representa la información asociada a un documento generado y/o incorporado a un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDOCEXP	TpoPK	Identificador del documento del expediente
→ TIPODOC	TrTipoDocumento	Tipo de documento
→ FECHA	Timestamp	Fecha de generación o incorporación
→ PRESENTADO	String	Para los documentos a "incorporar" indica si el documento se ha presentado o no. → " S " – El documento se ha presentado → " N " – El documento no se ha presentado
→ CORRECTO	String	Para los documentos a "incorporar" indica si el documento se ha presentado correcto o no. Esto no significa que no se pueda utilizar con los documentos generados. → " S " – El documento se ha presentado correcto → " N " – El documento no se ha presentado correctamente

→	FECHALIMITE	Timestamp	Fecha límite de presentación. (Sólo para los documentos a incorporar)
→	ESTADO	String	Estado en el que se encuentra el documento. (Este dato no tiene sentido para documentos incorporados) → “ R ” – El documento se encuentra en realización → “ E ” – El documento se encuentra pendiente de firma → “ F ” – El documento se encuentra firmado → “ D ” – El documento está descartado → “ T ” – El documento está terminado → “ V ” – El documento está versionado
→	OBSERVACIONES	String	Observaciones del documento
→	REFEXPXNAS	TpoPK	Identificador del paso en la evolución del expediente en el que se generó y/o incorporó.
→	USUARIO	String	Usuario que generó y/o incorporó el documento al expediente
→	FECHAFIRMA	Timestamp	Fecha de la última firma
→	REGISTRODOC	TrRegistroDocumento	Datos de registro que existen para un documento
→	FECHAFIRMADIGITAL	Timestamp	Fecha de la última firma digital.
→	FECHAFIN	Timestamp	Fecha de terminación, anulación, descarte o firma del documento., es decir del cambio de estado.
→	FECHAARCHIVO	Timestamp	Fecha de archivo del documento.
→	FECHACADUCIDAD	Timestamp	Fecha de caducidad del documento.
→	INFORMADO	String	Indica si se ha informado al bus del documento. → “ S ” – Si se ha informado → “ N ” – No se ha informado
→	REUTILIZABLE	String	Indica si el documento es reutilizable → “ S ” – Sí → “ N ” – No
→	FIRMADIG	String	Indica si el documento es firmable digitalmente. → “ S ” – Si es firmable digitalmente → “ N ” – No es firmable digitalmente
→	VERSION	String	Versión del documento.
→	REFDOCEXPVERS	TpoPK	Identificador del documento al que versiona
→	MODOGEN	String	Modo de generación del documento → “ R ” – Reports Server Oracle → “ P ” – Java PDF → “ O ” – OpenOffice Bean
→	REFFASE	TpoPK	Identificador de la fase
→	NOMBREFASE	String	Nombre de la fase

→	REFDEFPROC	TpoPK	Identificador de la definición del procedimiento
→	NOMBREUSU	String	Nombre y apellidos del usuario
→	CODHASH	String	Código de identificación de documento para Notific@, Port@firmas
→	REFEXPEDIENTE	TpoPK	Identificador del expediente
→	NUMEXP	String	Número del expediente
→	TITULOEXP	String	Título del expediente
→	OBSERVACIONES EXP	String	Observaciones del expediente
→	REFORGENVEXP	TpoPK	Identificador del organismo al que pertenece el expediente
→	REFORGENVEXP	TpoPK	Identificador del organismo que envía el expediente
→	REFWARDA	String	Identificador del documento en w@rdA
→	REFWARDAANX	String	Identificador del anexo del documento en w@rdA
→	REFCOMPONENTE	TpoPK	Identificador del componente en el que está guardado.
→	NOMBREFICHERO	String	Nombre del fichero del documento físico.
→	FORMATO	String	Tipo de formato del fichero del documento.
→	PROPIETARIOEXP	String	Usuario propietario del expediente
→	FECHASACUSENOTI	Timestamp[]	<p>Conjunto de fechas de acuse de las notificaciones del documento. Si el atributo es nulo indica que no existen notificaciones para el documento. Si el atributo es distinto de nulo su tamaño indica el número de notificaciones del documento. En cada una de las posiciones del array se encuentra la fecha de acuse en caso de que tenga, si no su valor será null.</p> <p>Por ejemplo: docExp.getFECHASACUSE().length == 4 → Indica que tiene 4 notificaciones docExp.getFECHASACUSE()[0] == 01/01/2010 docExp.getFECHASACUSE()[1] == 02/02/2010 docExp.getFECHASACUSE()[2] == null docExp.getFECHASACUSE()[3] == null</p> <p>En este caso el documento tiene 4 notificaciones de las cuáles 2 ya tienen fecha de acuse y otras 2 todavía no. Indica de qué clase es el documento.</p>
→	CLASEDOC	String	Indica de qué clase es el documento.

<p>→ CODHASHDOCORIG</p> <p>→ ALGORITMO</p> <p>→ DOCUMENTO</p> <p>→ FECHA_MODIFICACION</p> <p>→ IDENTIFICADORENI</p> <p>→ INFORMACIONPUBLICA</p> <p>→ NUMDOC</p> <p>→ NUMDOCORIGEN</p> <p>→ ORGANOENI</p> <p>→ REFESTADOELAB</p> <p>→ REFTIPDOCENI</p> <p>→ INFORMEFIRMA</p>	<p>String</p> <p>String</p> <p>byte[]</p> <p>Timestamp</p> <p>String</p> <p>String</p> <p>String</p> <p>String</p> <p>TpoPK</p> <p>TpoPK</p> <p>byte[]</p>	<p>Sus valores pueden ser: → “G” – General → “S” – SCSP con necesidad de consentimiento → “L” – SCSP autorizado por Ley Código hash del documento original</p> <p>Algoritmo hash para el código hash del documento original</p> <p>Contenido binario del documento</p> <p>Fecha de modificación</p> <p>Identificador único del documento en formato ENI</p> <p>Indica si el documento será visible para información pública. Sus valores pueden ser: → “S” – Es visible para información pública → “N” – No es visible para información pública Número de documento en la aplicación “cliente”</p> <p>Identificador del documento origen en formato ENI, en caso que sea una copia auténtica.</p> <p>Organismo que realizó la captura del documento</p> <p>Estados del documento. Pueden ser: → “O” – Original → “F” – Copia electrónica auténtica con cambio de formato → “D” – Copia electrónica auténtica de documento papel → “P” – Copia electrónica parcial auténtica → “T” – Otros Identificador del tipo documental eni.</p> <p>Contenido informe de firma</p>
---	--	--

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDOCEXP	NUMÉRICO	Identificador del documento del expediente
→ CAMPO_ETIQUETA	CADENA(11)	Abreviatura del tipo de documento
→ CAMPO_NOMBRE	CADENA(50)	Nombre del tipo de documento
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del tipo de documento
→ CAMPO_FECHA	DATE	Fecha de generación o incorporación
→ CAMPO_PRESENTADO	CADENA(1)	Para los documentos a “incorporar” indica si el documento se ha presentado o no. → “ S ” – El documento se ha presentado

→	CAMPO_CORRECTO	CADENA(2)	<p>→ “N” – El documento no se ha presentado Para los documentos a “incorporar” indica si el documento se ha presentado correcto o no. Esto no significa que no se pueda utilizar con los documentos generados.</p> <p>→ “S” – El documento se ha presentado correcto</p>
→	CAMPO_FECHALIMITE	DATE	<p>→ “N” – El documento no se ha presentado correctamente Fecha límite de presentación. (Sólo para los documentos a incorporar)</p>
→	CAMPO_ESTADO	CADENA(1)	<p>Estado en el que se encuentra el documento. (Este dato no tiene sentido para documentos incorporados)</p> <p>→ “R” – El documento se encuentra en realización</p> <p>→ “E” – El documento se encuentra pendiente de firma</p> <p>→ “F” – El documento se encuentra firmado</p> <p>→ “D” – El documento está descartado</p> <p>→ “T” – El documento está terminado</p> <p>→ “V” – El documento está versionado</p>
→	CAMPO_OBSERVACIONES	CADENA(250)	Observaciones del documento
→	CAMPO_ENTRADASALIDA	CADENA(2)	<p>Indica cómo está definido el documento en TREW@</p> <p>→ “E” – Entrada</p> <p>→ “S” – Salida</p> <p>→ “ES” – Entrada / Salida</p>
→	CAMPO_INCGEN	CADENA(1)	<p>Indica cómo está definido el documento</p> <p>→ “I” – Documento a Incorporar al expediente</p> <p>→ “G” – Documento a Generar</p>
→	CAMPO_USUARIO	CADENA(10)	Usuario que generó y/o incorporó el documento al expediente
→	CAMPO_FECHAFIRMA	DATE	Fecha de la última firma del documento
→	CAMPO_REFEXPFAS	NUMÉRICO	Identificador del paso en la evolución del expediente en el que se generó y/o incorporó.
→	CAMPO_FECHAFIRMADIGITAL	DATE	Fecha de la última firma digital
→	CAMPO_FECHAFIN	DATE	Fecha de terminación, anulación, descarte o firma del documento, es decir del cambio de estado
→	CAMPO_FECHAARCHIVO	DATE	Fecha de archivo del documento.
→	CAMPO_FECHACADUCIDAD	DATE	Fecha de caducidad del documento.
→	CAMPO_INFORMADO	CADENA(1)	<p>Indica si se ha informado al bus del documento.</p> <p>→ “S” – Si se ha informado</p> <p>→ “N” – No se ha informado</p>
→	CAMPO_VERSIONABLE	CADENA(1)	<p>Indica si el documento es versionable</p> <p>→ “S” – Sí</p> <p>→ “N” – No</p>
→	CAMPO_REGISTRABLE	CADENA(1)	<p>Indica si el documento es registrable</p> <p>→ “S” – Sí</p> <p>→ “N” – No</p>
→	CAMPO_NOTIFICABLE	CADENA(1)	<p>Indica si el documento es notificable</p> <p>→ “S” – Sí</p> <p>→ “N” – No</p>
→	CAMPO_REUTILIZABLE	CADENA(1)	<p>Indica si el documento es reutilizable</p> <p>→ “S” – Sí</p> <p>→ “N” – No</p>
→	CAMPO_FIRMADIG	CADENA(1)	<p>Indica si el documento es firmable digitalmente.</p> <p>→ “S” – Si es firmable digitalmente</p>



→	CAMPO_TIPOFIRMA	CADENA(1)	→ "N" – No es firmable digitalmente Tipo de firma para el documento → "C" – Firma en cascada → "P" – Firma en paralelo
→	CAMPO_VERSION	CADENA(9)	Versión del documento.
→	CAMPO_REFDOCEXPVERS	NUMERICO	Identificador del documento al que versiona
→	CAMPO_MULTIPLE	CADENA(1)	Indica si el tipo de documento es múltiple → "S" – Sí → "N" – No
→	CAMPO_MODALOGEN	CADENA(1)	Modo de generación del documento → "R" – Report Server Oracle → "P" – Generación PDF → "O" – OpenOffice Bean
→	CAMPO_TEXTOAUXILIAR	CADENA(50)	Texto auxiliar del tipo de documento
→	CAMPO_REFFASE	NUMERICO	Identificador de la fase
→	CAMPO_NOMBREFASE	CADENA(50)	Nombre de la fase
→	CAMPO_REFDEFPROC	NUMERICO	Identificador de la definición del procedimiento
→	CAMPO_NOMBREUSU	CADENA(98)	Nombre y apellidos del usuario
→	CAMPO_CODHASH	CADENA(20)	Código de identificación de documento para notific@, port@firma
→	CAMPO_REFTIPODOC	NUMERICO	Identificador del tipo de documento
→	CAMPO_REFEXPEDIENTE	NUMERICO	Identificador del expediente
→	CAMPO_NUMEXP	CADENA(30)	Número del expediente
→	CAMPO_TITULOEXP	CADENA(250)	Título del expediente
→	CAMPO_OBSERVACIONES EXP	CADENA(250)	Observaciones del expediente
→	CAMPO_REFORGENVEXP	NUMÉRICO	Identificador del organismo al que pertenece el expediente
→	CAMPO_REFORGENVEXP	NUMÉRICO	Identificador del organismo que envía el expediente
→	CAMPO_FECHAARCHIVO EXP	DATE	Fecha de archivo del expediente
→	CAMPO_ESTADOARCHIVA EXP	CADENA(1)	Estado de la petición de archivado Sus valores pueden ser: → "p" – Pendiente → "A" – Archivado → "R" – Rechazado
→	CAMPO_REFWARDA	CADENA(50)	Identificador del documento en w@rdA
→	CAMPO_REFWARDAANX	CADENA(50)	Identificador del anexo del documento en w@rdA
→	CAMPO_REFCOMPONENT E	NUMERICO	Identificador del componente en el que está guardado.
→	CAMPO_NOMBREFICHERO	CADENA(64)	Nombre del fichero del documento físico.
→	CAMPO_FORMATO	CADENA(128)	Formato del fichero del documento.
→	CAMPO_REFSISTEMA	NUMÉRICO	Identificador del sistema
→	CAMPO_CODSISTEMA	CADENA(10)	Código del sistema
→	CAMPO_DESCSISTEMA	CADENA(128)	Descripción del sistema
→	CAMPO_CONEXBUS	CADENA(1)	Indica si se tiene conexión al bus. → "S" – Sí → "N" – No → "O" – Opcional
→	CAMPO_REFCOMPINF	NUMÉRICO	Identificador del componente a informar.

→	CAMPO_REFCOMP AVI	NUMÉRICO	Identificador del componente del @visorador.
→	CAMPO_REFCOMP GUAR	NUMÉRICO	Identificador del componente que guarda los documentos.
→	CAMPO_REFCOMP ARCH	NUMÉRICO	Identificador del componente que archiva.
→	CAMPO_REFCOMP REG	NUMÉRICO	Identificador del componente que registra los documentos.
→	CAMPO_REFCOMP FIRMA	NUMÉRICO	Identificador del componente que firma los documentos.
→	CAMPO_REFCOMP NOTI	NUMÉRICO	Identificador del componente que notifica los documentos.
→	CAMPO_PROPIETARIO EXP	CADENA(10)	Usuario propietario del expediente
→	CAMPO_FECHAS ACUSE NOTI	DATE	Conjunto de fechas de acuse de las notificaciones del documento. Al hacer uso de este filtro sólo se devuelven los documentos que tengan notificaciones y de éstas sólo las que cumplan con el filtro indicado para la fecha de acuse. Por ejemplo si se filtra para que la fecha de acuse sea null nos devolverá los documentos que tengan notificaciones y en el conjunto de las notificaciones sólo vendrán aquellas que sean null, cómo indica el filtro.
→	CAMPO_CLASE DOCTIPO DOC	CADENA(20)	Indica de qué clase es el tipo de documento. Sus valores pueden ser: → “ G ” – General → “ S ” – SCSP con necesidad de consentimiento → “ L ” – SCSP autorizado por Ley
→	CAMPO_CLASE DOC	CADENA(20)	Indica de qué clase es el documento. Sus valores pueden ser: → “ G ” – General → “ S ” – SCSP con necesidad de consentimiento → “ L ” – SCSP autorizado por Ley
→	CAMPO_COD HASH DOCTIPO ORIGINAL	CADENA(50)	Código hash del documento original
→	CAMPO_ALGORITMO HASH	CADENA(10)	Algoritmo hash para el código hash del documento original
→	CAMPO_ADMINISTRATIVO		
→	CAMPO_ASUNTO		
→	CAMPO_DESTINATARIO		
→	CAMPO_ESTADO ELABENI	NUMÉRICO	Estado elaboración de documento ENI
→	CAMPO_FECHA ENTRADA		
→	CAMPO_FECHA SALIDA		
→	CAMPO_IDENTIFICADOR ENI	CADENA(55)	Identificador del documento electrónico ENI. Garantiza la unicidad de los documentos a nivel estatal.
→	CAMPO_INFORMACION PÚBLICA	char(1)	Indica si el documento será visible para información pública. Sus valores pueden ser: → “ S ” – El documento será visible → “ N ” – El documento no será visible
→	CAMPO_INFORMAR		
→	CAMPO_NUM DOC	CADENA(60)	Número de documento en la aplicación “cliente”
→	CAMPO_NUM DOC ORIGEN	CADENA(50)	Identificador del documento origen en formato ENI, en caso que sea una copia auténtica.
→	CAMPO_NUM REGENTRADA		
→	CAMPO_NUM REGSALIDA		
→	CAMPO_OBSERVACIONES EXP		
→	CAMPO_OFICINA ENTRADA		

- **CAMPO_OFICINASALIDA**
- **CAMPO_ORGANOENI** CADENA(10) Organismo que realizo la captura del documento.
- **CAMPO_REMITENTE**

Métodos para aplicación de los filtros

- importarDocumento
- incorporarDocExternoMultiplesExpedientes
- incorporarDocumento
- incorporarDocumentoExterno
- incorporarDocumentoMultiplesExpedientes
- incorporarDocumentoNoDefinido
- incorporarDocumentoNoDefinidoExterno
- modificarDatosDocumento
- obtenerDatosFichero
- obtenerDocExp
- obtenerDocumentosExpediente
- obtenerTareaDocumentoExpModvale

Campos definidos para las exclusiones

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ EXCL_FECHAFIRMA	Excluye los datos de la fecha de firma de la consulta
→ EXCL_EXPFASE	Excluye los datos de la entidad expedientes en fase y fase de la consulta. Si se excluyen los datos de expediente en fase no se tiene en cuenta el parámetro soloUsuario.
→ EXCL_NOMBREUSU	Excluye los datos de nombre de usuario de la consulta
→ EXCL_SISTEMA	Excluye los datos de la entidad sistema de la consulta
→ EXCL_EXPEDIENTE	Excluye los datos de la entidad expediente de la consulta
→ EXCL_FECHASACUSENOTI	Excluye las fechas de las notificaciones de los documentos

Métodos para aplicación de exclusiones

- obtenerDocumentosExpediente

2.4.1.2.29 TrDocumentoExpedienteSimple

trewa.bd.trapi.trapiui.tpo.TrDocumentoExpedienteSimple

Clase que representa la información asociada a un documento generado y/o incorporado a un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDOCEXP	TpoPK	Identificador del documento del expediente
→ TIPODOC	TrTipoDocumento	Tipo de documento
→ FECHA	Timestamp	Fecha de generación o incorporación
→ PRESENTADO	String	Para los documentos a "incorporar" indica si el documento se ha presentado o no. → " S " – El documento se ha presentado → " N " – El documento no se ha presentado
→ CORRECTO	String	Para los documentos a "incorporar" indica si el documento se ha presentado correcto o no. Esto no significa que no se pueda utilizar con los documentos generados. → " S " – El documento se ha presentado correcto → " N " – El documento no se ha presentado correctamente
→ FECHALIMITE	Timestamp	Fecha límite de presentación. (Sólo para los documentos a incorporar)
→ ESTADO	String	Estado en el que se encuentra el documento. (Este dato no tiene sentido para documentos incorporados) → " R " – El documento se encuentra en realización → " E " – El documento se encuentra pendiente de firma → " F " – El documento se encuentra firmado → " D " – El documento está descartado → " T " – El documento está terminado → " V " – El documento está versionado
→ OBSERVACIONES	String	Observaciones del documento
→ REFEXPXNAS	TpoPK	Identificador del paso en la evolución del expediente en el que se generó y/o incorporó.
→ USUARIO	String	Usuario que generó y/o incorporó el documento al expediente
→ FECHAFIN	Timestamp	Fecha de terminación, anulación, descarte o firma del documento., es decir del cambio de estado.
→ FECHAARCHIVO	Timestamp	Fecha de archivo del documento.
→ FECHACADUCIDA	Timestamp	Fecha de caducidad del documento.
→ D INFORMADO	String	Indica si se ha informado al bus del documento. → " S " – Si se ha informado → " N " – No se ha informado
→ REUTILIZABLE	String	Indica si el documento es reutilizable → " S " – Sí → " N " – No
→ FIRMADIG	String	Indica si el documento es firmable digitalmente. → " S " – Si es firmable digitalmente → " N " – No es firmable digitalmente
→ VERSION	String	Versión del documento.
→ REFDOCEXPVERS	TpoPK	Identificador del documento al que versiona
→ MODOGEN	String	Modo de generación del documento → " R " – Reports Server Oracle → " P " – Java PDF

→	CODHASH	String	→ “O” – OpenOffice Bean Código de identificación de documento para notific@, port@firma
→	REFEXPEDIENTE	TpoPK	Identificador del expediente
→	REFWARDA	String	Identificador del documento en w@rdA
→	REFWARDAAANX	String	Identificador del anexo del documento en w@rdA
→	REFCOMPONENTE	TpoPK	Identificador del componente en el que está guardado.
→	NOMBREFICHERO	String	Nombre del fichero del documento físico.
→	FORMATO	String	Tipo de formato del fichero del documento.
→	ALGORITMOHASH	String	Algoritmo hash del documento original
→	CODHASHDOCORI	String	Código hash del documento original

G

Campos definidos para filtrados y ordenación

	<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_REFDOCEXP	NUMÉRICO	Identificador del documento del expediente
→	CAMPO_ETIQUETA	CADENA(11)	Abreviatura del tipo de documento
→	CAMPO_NOMBRE	CADENA(50)	Nombre del tipo de documento
→	CAMPO_DESCRIPCION	CADENA(250)	Descripción del tipo de documento
→	CAMPO_FECHA	DATE	Fecha de generación o incorporación
→	CAMPO_PRESENTADO	CADENA(1)	Para los documentos a “incorporar” indica si el documento se ha presentado o no. → “S” – El documento se ha presentado → “N” – El documento no se ha presentado
→	CAMPO_CORRECTO	CADENA(2)	Para los documentos a “incorporar” indica si el documento se ha presentado correcto o no. Esto no significa que no se pueda utilizar con los documentos generados. → “S” – El documento se ha presentado correcto → “N” – El documento no se ha presentado correctamente
→	CAMPO_FECHALIMITE	DATE	Fecha límite de presentación. (Sólo para los documentos a incorporar)
→	CAMPO_ESTADO	CADENA(1)	Estado en el que se encuentra el documento. (Este dato no tiene sentido para documentos incorporados) → “R” – El documento se encuentra en realización → “E” – El documento se encuentra pendiente de firma → “F” – El documento se encuentra firmado → “D” – El documento está descartado → “T” – El documento está terminado → “V” – El documento está versionado
→	CAMPO_OBSERVACIONES	CADENA(250)	Observaciones del documento
→	CAMPO_ENTRADASALIDA	CADENA(2)	Indica cómo está definido el documento en TREW@ → “E” – Entrada → “S” – Salida → “ES” – Entrada / Salida
→	CAMPO_INCGEN	CADENA(1)	Indica cómo está definido el documento → “I” – Documento a Incorporar al expediente → “G” – Documento a Generar

→	CAMPO_USUARIO	CADENA(10)	Usuario que generó y/o incorporó el documento al expediente
→	CAMPO_FECHAFIN	DATE	Fecha de terminación, anulación, descarte o firma del documento, es decir del cambio de estado
→	CAMPO_FECHAARCHIVO	DATE	Fecha de archivo del documento.
→	CAMPO_FECHACADUCIDA	DATE	Fecha de caducidad del documento.
	D		
→	CAMPO_INFORMADO	CADENA(1)	Indica si se ha informado al bus del documento. → “ S ” – Si se ha informado → “ N ” – No se ha informado
→	CAMPO_VERSIONABLE	CADENA(1)	Indica si el documento es versionable → “ S ” – Sí → “ N ” – No
→	CAMPO_REGISTRABLE	CADENA(1)	Indica si el documento es registrable → “ S ” – Sí → “ N ” – No
→	CAMPO_NOTIFICABLE	CADENA(1)	Indica si el documento es notificable → “ S ” – Sí → “ N ” – No
→	CAMPO_REUTILIZABLE	CADENA(1)	Indica si el documento es reutilizable → “ S ” – Sí → “ N ” – No
→	CAMPO_FIRMADIG	CADENA(1)	Indica si el documento es firmable digitalmente. → “ S ” – Si es firmable digitalmente → “ N ” – No es firmable digitalmente
→	CAMPO_TIPOFIRMA	CADENA(1)	Tipo de firma para el documento → “ C ” – Firma en cascada → “ P ” – Firma en paralelo
→	CAMPO_VERSION	CADENA(9)	Versión del documento.
→	CAMPO_REFDOCEXPVERS	NUMERICO	Identificador del documento al que versiona
→	CAMPO_MULTIPLE	CADENA(1)	Indica si el tipo de documento es múltiple → “ S ” – Sí → “ N ” – No
→	CAMPO_MODALGEN	CADENA(1)	Modo de generación del documento → “ R ” – Report Server Oracle → “ P ” – Generación PDF → “ O ” – OpenOffice Bean
→	CAMPO_TEXTOAUXILIAR	CADENA(50)	Texto auxiliar del tipo de documento
→	CAMPO_CODHASH	CADENA(20)	Código de identificación de documento para notific@, port@firma
→	CAMPO_REFTIPODOC	NUMERICO	Identificador del tipo de documento
→	CAMPO_REFEXPEDIENTE	NUMERICO	Identificador del expediente
→	CAMPO_REFWARDA	CADENA(50)	Identificador del documento en w@rdA
→	CAMPO_REFWARDAANX	CADENA(50)	Identificador del anexo del documento en w@rdA
→	CAMPO_REFCOMPONENT	NUMERICO	Identificador del componente en el que está guardado.
	E		
→	CAMPO_NOMBREFICHERO	CADENA(64)	Nombre del fichero del documento físico.
→	CAMPO_FORMATO	CADENA(128)	Formato del fichero del documento.
→	CAMPO_REFSISTEMA	NUMÉRICO	Identificador del sistema
→	CAMPO_ADMINISTRATIVO	CADENA()	Indica si el documento es administrativo o auxiliar
→	CAMPO_ALGORITMOHASH	CADENA(10)	Algoritmo hash del documento original

- **CAMPO_CODHASHDOCORIG** CADENA(50) Código hash del documento original
- **CAMPO_EDITABLE** CADENA() Indica si el documento es editable
- **CAMPO_REFEXPFAS** NUMÉRICO Identificador del paso en la evolución del expediente en el que se generó y/o incorporó

Métodos para aplicación de los filtros

→ obtenerDocumentosExpedienteSimple

2.4.1.2.30 TrDocumentoExplorador

trewa.bd.trapi.trapiui.tpo.TrDocumentoExplorador

Clase que representa la información referente a las tareas pendientes de tipo "GENERAR_DOCUMENTO" o "INCORPORAR_DOCUMENTO" definidas en una fase de un procedimiento.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	REFDOCPER	TpoPK	Identificador del documento permitido
→	OBLIGATORIO	String	Indica si el documento es obligatorio generarlo o incorporarlo. → " S " – El documento es obligatorio → " N " – El documento no es obligatorio
→	DESCRIPCION	String	Descripción del documento permitido
→	ORDEN	Integer	Orden del documento permitido

2.4.1.2.31 TrDocumentoPermitido

trewa.bd.trapi.trapiui.tpo.TrDocumentoPermitido

Clase que representa la información referente a las tareas de tipo "GENERAR_DOCUMENTO" o "INCORPORAR_DOCUMENTO" definidas en una fase de un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDOCPER	TpoPK	Identificador del documento permitido
→ TIPODOC	TrTipoDocumento	Tipo de documento
→ OBLIGATORIO	String	Indica si el documento es obligatorio generarlo o incorporarlo. → “ S ” – El documento es obligatorio → “ N ” – El documento no es obligatorio
→ PERMISO	String	Permiso que tiene el usuario sobre el documento → “ G ” – Permiso para generar → “ E ” – Permiso para editar → “ F ” – Permiso para firmar → “ I ” – Permiso para incorporar → “ T ” – Todos
→ INFORMAR	String	Indica si se informa al bus del documento → “ S ” – Sí → “ N ” – No
→ ETIQUETA	String	Texto con el que se etiqueta el documento permitido en la fase
→ DESCRIPCION	String	Descripción del documento permitido
→ ETIQLARGA	String	Etiqueta larga para el documento permitido
→ PLAZO	TrPlazo	Plazo de realización
→ ORDEN	Integer	Orden del documento permitido
→ TIPOACTO	TrTipoActo	Tipo de acto
→ TEXTOAUXILIAR	String	Texto auxiliar del documento

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDOCPER	NUMÉRICO	Identificador del documento permitido
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento
→ CAMPO_ETIQUETA	CADENA(11)	Abreviatura del tipo de documento
→ CAMPO_NOMBRE	CADENA(50)	Nombre del tipo de documento
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del tipo de documento
→ CAMPO_OBLIGATORIO	CADENA(1)	Indica si el documento es obligatorio generarlo o incorporarlo. → “ S ” – El documento es obligatorio → “ N ” – El documento no es obligatorio
→ CAMPO_ENTRADASALIDA	CADENA(2)	Indica como está definido el documento en Trew@ → “ E ” – Entrada → “ S ” – Salida → “ ES ” – Entrada/Salida
→ CAMPO_INCGEN	CADENA(1)	Indica como está definido el documento. → “ I ” – Documento a incorporar al expediente → “ G ” – Documento a generar
→ CAMPO_PERMISO	CADENA(1)	Permiso que tiene el usuario sobre el documento.

			→ "G" – Permiso para generar → "E" – Permiso para editar → "F" – Permiso para firmar → "I" – Permiso para incorporar → "T" – Todos
→	CAMPO_MULTIPLE	CADENA(1)	Indica si el tipo de documento es múltiple o no. → "S" – Tipo de documento es múltiple → "N" – Tipo de documento no es múltiple
→	CAMPO_TEXTOAUXILIAR	CADENA(50)	Texto auxiliar del tipo de documento
→	CAMPO_CLASEDOC	CADENA(1)	Indica de que clase es el documento. Por ejemplo SCSP o GENERAL
→	CAMPO_INFORMAR	CADENA(1)	Indica si se informa al bus del documento. Sus valores pueden ser: → "S" – Sí → "N" – No
→	CAMPO_ETIQUETADOC	CADENA(11)	Texto con el que se etiqueta al documento permitido en la fase
→	CAMPO_DESCRIPCIONDOC	CADENA(250)	Descripción de lo que representa el documento permitido en la fase
→	CAMPO_ETIQLARGA	CADENA(25)	Etiqueta larga para el documento permitido
→	CAMPO_UNIDAD	CADENA(1)	Unidad de medida para el plazo máximo de realización. Sus valores pueden ser: → "D" – Días → "M" – Meses → "A" – Años
→	CAMPO_NUMUNIDADES	CADENA(2)	Indica el número de unidades del plazo máximo de realización número de días, número de meses, número de años
→	CAMPO_DESCFECHALIMIT	CADENA(100)	Descripción del significado de la fecha límite de realización de la tarea
→	CAMPO_ORDEN	NUMÉRICO	Orden
→	CAMPO_TEXTOAUXILIARD	CADENA(50)	Texto auxiliar del documento permitido
→	CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto
→	CAMPO_ABREVTIPOACTO	CADENA(10)	Abreviatura del tipo de acto
→	CAMPO_DESCTIPOACTO	CADENA(200)	Descripción del tipo de acto
→	CAMPO_REFSTMATIPOACTO	NUMÉRICO	Identificador del sistema del tipo de acto

Métodos para aplicación de los filtros

→obtenerDocumentosPermitidos

2.4.1.2.32 TrEmpleado

trewa.bd.trapi.trapiui.tpo.TrEmpleado

Clase que representa la información referente a los usuarios que ocupan los puestos de trabajo en los organismos definidos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoria
→ USUARIO	TrUsuario	Usuario
→ ORGANISMO	TrOrganismo	Organismo
→ PTOTRABAJO	TrPuestoTrabajo	Puesto de trabajo
→ TIPO	String	Tipo de empleado → "F" – Funcionario → "L" – Laboral
→ FECHANOMBRAMIENTO	Timestamp	Fecha de ocupación del puesto por el usuario.
→ FECHACESE	Timestamp	Fecha de baja del usuario en el puesto
→ TRATAMIENTO	String	Texto de tratamiento para los documentos (D., D ^a , Ilmo., etc.)
→ TXTPREFERENCIA	String	Texto de referencia para el usuario.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_USUARIO	CADENA(10)	Código del usuario
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo
→ CAMPO_ORGANISMO	CADENA(32)	Nombre del organismo
→ CAMPO_CODPTOTRAB	CADENA(10)	Código del puesto de trabajo
→ CAMPO_PTOTRABAJO	CADENA(32)	Nombre del puesto de trabajo
→ CAMPO_TIPO	CADENA(1)	Tipo de empleado → "F" – Funcionario → "L" – Laboral
→ CAMPO_FECHANOMBRAMIENTO	DATE	Fecha de ocupación del puesto por el usuario.
→ CAMPO_FECHACESE	DATE	Fecha de baja del usuario en el puesto
→ CAMPO_TRATAMIENTO	CADENA(64)	Texto de tratamiento para los documentos (D., D ^a , Ilmo., etc.)
→ CAMPO_TXTPREFERENCIA	CADENA(10)	Texto de referencia para el usuario.
→ CAMPO_CODMUNICIPIOORG	CADENA(3)	Código del municipio del organismo
→ CAMPO_CODPROVINCIAORG	CADENA(2)	Código de la provincia del organismo
→ CAMPO_TIPOVIAORG	CADENA(10)	Tipo de vía del organismo
→ CAMPO_CODORG	CADENA(16)	Código del organismo
→ CAMPO_NOMBREORG	CADENA(32)	Nombre del organismo



→	CAMPO_DESCRIPCIONORG	CADENA(200)	Descripción del organismo
→	CAMPO_TIPOORG	CADENA(2)	Código del tipo de organismo Sus valores pueden ser: <ul style="list-style-type: none">- "DC" – Delegación Provincial- "SC" – Servicios Centrales- "CE" – Centro Adscrito- "EE" – Empresa Externa
→	CAMPO_NOMBREAORG	CADENA(64)	Nombre de la vía pública del domicilio del organismo
→	CAMPO_NUMEROORG	NUMÉRICO	Número del domicilio del organismo
→	CAMPO_LETRAORG	CADENA(2)	Letra del domicilio del organismo
→	CAMPO_ESCALERAORG	CADENA(2)	Escalera del domicilio del organismo
→	CAMPO_PISOORG	NUMÉRICO	Piso del domicilio del organismo
→	CAMPO_PUERTAORG	CADENA(2)	Puerta del domicilio del organismo
→	CAMPO_CAMPO_CODPOSTALORG	NUMÉRICO	Código postal del domicilio del organismo
→	CAMPO_TELEFONOORG	CADENA(25)	Número(s) de teléfono de contacto del organismo.
→	CAMPO_TLFMOVILORG	CADENA(25)	Número(s) de teléfono móvil de contacto del organismo.
→	CAMPO_FAXORG	CADENA(25)	Número(s) de Fax del organismo.
→	CAMPO_EMAILORG	CADENA(64)	Correo electrónico del organismo
→	CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA.
→	CAMPO_IDARIESORG	CADENA(16)	Identificador de aries del organismo.
→	CAMPO_IDENTIFICADORORG	CADENA(15)	Número de identificación (CIF) del organismo.
→	CAMPO_DIGCONTROLORG	CADENA(1)	Dígito de control para el número de identificación del organismo.
→	CAMPO_FECHAINIVIGORG	DATE	Fecha de inicio de vigencia del organismo.
→	CAMPO_FECHAFINIVIGORG	DATE	Fecha de fin de vigencia del organismo
→	CAMPO_REFORGPADRE	NUMÉRICO	Identificador del organismo superior al que pertenece.
→	CAMPO_REFTIPOORG	NUMÉRICO	Identificador del tipo de organismo.
→	CAMPO_REFTIPOORGZ	NUMÉRICO	Identificador del tipo de organización del organismo.
→	CAMPO_ORDENPRELACIONORG	CADENA(15)	Orden de relación del organismo
→	CAMPO_CREADO	CADENA	Usuario que crea el registro
→	CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→	CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→	CAMPO_FECHAMODIFICADO	DATE	Fecha de modificación del registro
→	CAMPO_NOMBREUSUARIO	CADENA(32)	Nombre del usuario
→	CAMPO_APELLIDO1USUARIO	CADENA(32)	Primer apellido del usuario
→	CAMPO_APELLIDO2USUARIO	CADENA(32)	Segundo apellido del usuario
→	CAMPO_IDENTIFICADORUSUARIO	CADENA(15)	Identificador del usuario (Número del DNI por ejemplo)
→	CAMPO_TIPOIDENTUSUARIO	CADENA(1)	Tipo de identificador del usuario (DNI, NIF, ...)
→	CAMPO_SEXOUSUARIO	CADENA(1)	Sexo del usuario
→	CAMPO_FECHAALTAUSUARIO	DATE	Fecha de alta del usuario
→	CAMPO_FECHABAJAUSUARIO	DATE	Fecha de baja del usuario
→	CAMPO_EMAILUSUARIO	CADENA(64)	Correo electrónico del usuario
→	CAMPO_ANAGRAMAFISCALUSUARIO	CADENA(50)	Anagrama fiscal del usuario

Métodos para aplicación de los filtros

→ obtenerEmpleado

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.1.2.33 TrEnviarA

trewa.bd.trapi.trapiui.tpo.TrEnviarA

Clase que representa la información referente a un paso para llevar a un expediente a una situación concreta.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTRANSICION	TpoPK	Identificador de transición
→ REFFASE	TpoPK	Identificador de fase en la que se quedaría el expediente
→ REFFASEPADRE	TpoPK	Identificador de fase de la que parte si es un flujo reutilizable

2.4.1.2.34 TrEvolucionExpediente

trewa.bd.trapi.trapiui.tpo.TrEvolucionExpediente

Clase que representa la información referente a situaciones por las que pasa un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoria
→ REFEXPXFAS	TpoPK	Identificador del paso en la evolución

→	FASE	TrFase	Fase o situación
→	FECHAENTRADA	Timestamp	Fecha en la que se entró en la fase
→	FECHASALIDA	Timestamp	Fecha en la que se salió de la fase
→	FECHALIMITE	Timestamp	Fecha límite de salida de la fase
→	USUARIO	String	Usuario que lo dejó en la fase
→	USUSARIOBLQ	String	Usuario que lo tiene bloqueado en la fase
→	OBSERVACIONES	String	Observaciones del usuario que lo dejó en la fase
→	REFEXPXFASPADRE	TpoPK	Para los pasos pertenecientes a módulos reutilizables, indica el paso que dejó al expediente en la fase que representa a dicho módulo.
→	TRANSICION	TrTransicion	Transición
→	INFORMADABUS	String	Indica si se informa al bus de la entrada en la fase → “ S ” – Sí → “ N ” – No
→	INFORMADAFINBUS	String	Indica si se informa al bus de la salida de la fase → “ S ” – Sí → “ N ” – No
→	REFFASEPADRE	TpoPK	Identificador de la fase padre
→	NOMBREUSU	String	Nombre y apellidos del usuario
→	NOMBREUSUBLQ	String	Nombre y apellidos del usuario que bloquea
→	REFDEFPROC	TpoPK	Identificador de la definición del procedimiento
→	ABIERTAEVENTO	String	Indica si la fase viene propagada de un evento → “ S ” – Sí → “ N ” – No

Campos definidos para filtrados y ordenación

	<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_REFEXPXFAS	NUMÉRICO	Identificador del paso en la evolución
→	CAMPO_REFFASE	NUMÉRICO	Identificador de la fase o situación
→	CAMPO_FASE	CADENA(50)	Nombre de la fase
→	CAMPO_METAFASE	CADENA(50)	Nombre de la metafase
→	CAMPO_FECHAENTRADA	DATE	Fecha en la que se entró en la fase
→	CAMPO_FECHASALIDA	DATE	Fecha en la que se salió de la fase
→	CAMPO_FECHALIMITE	DATE	Fecha límite de salida de la fase
→	CAMPO_USUARIO	CADENA(10)	Usuario que lo dejó en la fase
→	CAMPO_USUARIOBLQ	CADENA(10)	Usuario que lo tiene bloqueado en la fase
→	CAMPO_OBSERVACIONES	CADENA(250)	Observaciones del usuario que lo dejó en la fase
→	CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición
→	CAMPO_DESCRANSICION	CADENA(50)	Descripción de la transición
→	CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto administrativo que representa la transición
→	CAMPO_REFEXPXFASPADRE	NUMÉRICO	Para los pasos pertenecientes a módulos reutilizables, indica el paso que dejó al expediente en la fase que representa a dicho módulo.

→	CAMPO_TIPOTRANS	CADENA(2)	Indica el tipo de la transición → “ N ” – Transición normal → “ D ” – Transición de división → “ U ” – Transición de unión → “ ES ” – Evento que provoca salida → “ EN ” – Evento que no provoca la salida
→	CAMPO_REFFASEPADRE	NUMÉRICO	Identificador de la fase padre
→	CAMPO_DESCFASE	CADENA(250)	Descripción de la fase
→	CAMPO_NOMBREUSU	CADENA(98)	Nombre y apellidos del usuario
→	CAMPO_NOMBREUSU BLQ	CADENA(98)	Nombre y apellidos del usuario que bloquea
→	CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→	CAMPO_ABIERTAERVE NTO	CADENA(1)	Indica si la fase viene propagada de un evento → “ S ” – Sí → “ N ” – No
→	CAMPO_INFORMADAB US	CADENA(1)	Indica si se informa al bus de la entrada en la fase → “ S ” – Sí → “ N ” – No
→	CAMPO_INFORMADAFI NBUS	CADENA(1)	Indica si se informa al bus de la salida de la fase → “ S ” – Sí → “ N ” – No
→	CAMPO_TEXTOAUXILI AR	CADENA(500)	Texto auxiliar de la fase
→	CAMPO_CREADO	CADENA	Usuario que crea el registro
→	CAMPO_FECHACREAD O	DATE	Fecha de creación del registro
→	CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→	CAMPO_FECHAMODIFI CA	DATE	Fecha de modificación del registro
→	CAMPO_REFMETAFAS E	DATE	Fecha de modificación del registro

Métodos para aplicación de los filtros

- obtenerEvolucionExpediente
- evniarExpedienteA
- obtenerEvolucionExpedienteCount
- tramitarExpedienteAux

2.4.1.2.35 TrExpediente

trewa.bd.trapi.trapiui.tpo.TrExpediente

Clase que representa la información referente a un expediente.

El modelo de referencia w@nda define un expediente como “cada materialización de un determinado procedimiento”.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	AUDITORIA	TrAuditoria	Datos de auditoria
→	REFEXP	TpoPK	Identificador del expediente
→	STMA	TrSistema	Sistema
→	TIPOEXP	TrTipoExpediente	Tipo de expediente
→	DEFPROC	TrDefProcedimiento	Definición del procedimiento
→	FECHAALTA	Timestamp	Fecha de alta
→	NUMEXP	String	Número del expediente en la aplicación "cliente"
→	TITULOEXP	String	Título del expediente en la aplicación "cliente"
→	OBSERVACIONES	String	Observaciones
→	ORGANISMO	TrOrganismo	Organismo al que pertenece
→	ORGENVIA	TrOrganismo	Organismo que lo ha enviado
→	OTROSDATOS	String	Indica si tiene otros datos → " S " – Sí → " N " – No
→	FECHAARCHIVO	Timestamp	Fecha de archivo del expediente
→	REFARCHIVA	String	Identificador de la petición de archivado
→	ESTADOARCHIVA	String	Estado de la petición de archivado Sus valores pueden ser: → " p " – Pendiente → " A " – Archivado → " R " – Rechazado
→	NUMREGARCHVIA	String	Número de registro del expediente archivado
→	INFORMADO	String	Indica si se ha informado del expediente al bus → " S " – Sí → " N " – No
→	REFCOMPONENTE	TpoPK	Identificador del componente en el que está archivado
→	URLWANDA	String	Url del expediente en w@ndA
→	PROPIETARIO	String	Usuario propietario del expediente
→	IDENTIFICADORENI	String	Identificador del expediente ENI
→	ORGANISMOENI	String	Organismo al que pertenece el expediente ENI
→	CLASIFICACION	String	Clasificación del expediente ENI
→	FECHA_ENTRADA_FASE	Timestamp	Fecha de la fase actual
→	DESCRIPCION_FASE	String	Descripción de la fase actual
→	CODIGO_REGISTRO	String	Código de registro
→	FECHA_REGISTRO	Timestamp	Fecha de registro
→	CSV_EXP	String	CSV del expediente ENI
→	INI_CIUDADANO	Boolean	Indica si el expediente fue iniciado por el ciudadano → True – El expediente fue iniciado por el ciudadano → False – El expediente fue iniciado por la administración

→ **ESTADO** String Establece el estado del expediente (E01: Abierto, E02: Finalizado)

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_ABBREVDEFPRO	CADENA(10)	Abreviatura de la definición de procedimiento
→ CAMPO_ABBREVTIPOEXPER	CADENA(10)	Abreviatura del tipo de expediente
→ CAMPO_CODORGANISMO	NUMÉRICO	Código del organismo al que pertenece el expediente
→ CAMPO_CODORGENVIA	NUMÉRICO	Código del organismo que envía
→ CAMPO_CODSTMADEFPROC	NUMÉRICO	Código del sistema de la definición de procedimiento
→ CAMPO_CODWANDARG	NUMÉRICO	Código Wanda del organismo
→ CAMPO_CODWANDARGENVIA	NUMÉRICO	Código Wanda del organismo que envía
→ CAMPO_CREADO	CADENA(10)	Usuario que crea el registro
→ CAMPO_DESCDEFPROC	CADENA(50)	Descripción del procedimiento
→ CAMPO_DESCORGANISMO	CADENA(50)	Descripción del organismo
→ CAMPO_DESCORGENVIA	CADENA(50)	Descripción del organismo que envía
→ CAMPO_DESCTIPOEXPER	CADENA(50)	Descripción del tipo expediente
→ CAMPO_FECHAALTA	DATE	Fecha de alta del expediente
→ CAMPO_FECHAARCHIVO	DATE	Fecha de archivo del expediente
→ CAMPO_FECHACREADO	DATE	Fecha de creación del expediente
→ CAMPO_FECHAMODIFICADO	DATE	Fecha de modificación del expediente
→ CAMPO_INFORMADO	CADENA(1)	Indica si se ha informado al bus del expediente Sus valores pueden ser: → "S" – Sí → "N" – No
→ CAMPO_MODIFICADO	CADENA(10)	Usuario que modifica el registro
→ CAMPO_NOMBREORGANISMO	CADENA(10)	Nombre del organismo
→ CAMPO_NOMBREORGANISMOENVIA	CADENA(10)	Nombre del organismo que envía
→ CAMPO_NUMEXP	CADENA(10)	Número de expediente
→ CAMPO_OBSERVACIONES	CADENA(10)	Observaciones del expediente

	NES		
→	CAMPO_ORDENPRELACIONORG	CADENA(10)	Orden de relación del organismo
→	CAMPO_OTROSDATOS	CADENA(1)	Indica si el expediente tiene otros datos Sus valores pueden ser: → "S" – Sí → "N" – No
→	CAMPO_PROPIETARIO	CADENA	Usuario propietario del expediente
→	CAMPO_REFCOMPONENTE	NUMERICO	Identificador del componente donde está registrado el expediente
→	CAMPO_REFDEFPROC	NUMERICO	Identificador de la definición de procedimiento
→	CAMPO_REFEXP	NUMERICO	Referencia del expediente
→	CAMPO_REFORGANISMO	NUMERICO	Referencia del organismo
→	CAMPO_REFORGENVIA	NUMERICO	Referencia del organismo que envía
→	CAMPO_REFSTMADEFPROC	NUMERICO	Referencia del sistema de la definición de procedimiento
→	CAMPO_REFTIPOEXP	NUMERICO	Referencia del tipo expediente
→	CAMPO_TITULOEXP	NUMERICO	Título del expediente
→	CAMPO_URLWANDA	CADENA	URL w@nda
→	CAMPO_ESTADOARCHIVA	CADENA(1)	Estado de la petición de archivado Sus valores pueden ser: → "P" – Pendiente → "A" – Archivado → "R" – Rechazado
→	MAX_ROW	NUMERICO	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→	INIT_ROW	NUMERICO	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.
→	CAMPO_REFARCHIVA	CADENA(50)	Referencia de solicitud de ingreso en @rchivA
→	CAMPO_NUMREGARCHIVA	CADENA(50)	Número de registro de solicitud de ingreso en @rchivA
→	CAMPO_IDENTENI	CADENA(55)	Identificador de expediente ENI
→	CAMPO_CSV_EXP	CADENA(30)	CSV de expediente ENI
→	CAMPO_ORGANISMOENI	CADENA(10)	Organismo que abre el expediente
→	CAMPO_CLASIFICACIONENI	CADENA(50)	Clasificación del expediente ENI
→	CAMPO_QAA_PROC	CADENA(1)	Nivel de seguridad de acceso del procedimiento
→	CAMPO_NIVEL_INTRACCIONPROC	CADENA(1)	Nivel de integración con Carpeta Ciudadana del procedimiento
→	CAMPO_INICIUDADANO	CADENA(1)	Iniciado por el ciudadano
→	CAMPO_ORDENPRELACIONORGENVIA	CADENA(15)	Orden de relación del organismo que envía el expediente

Campos definidos para las exclusiones

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ EXCL_SISTEMA	Excluye los datos de la entidad sistema de la consulta
→ EXCL_ORGENVIA	Excluye los datos de la entidad organismo que envía de la consulta
→ EXCL_ORGANISMO	Excluye los datos de la entidad organismo de la consulta

Métodos para aplicación de los filtros y su uso

- crearExpediente
- explorarExpedientesSimple
- explorarExpedientes
- importarExpediente
- importarExpedienteEni
- obtenerExpedientes
- obtenerExpedientesCaducados
- obtenerExpedientesEnFase
- obtenerExpedientesPendientes
- obtenerExpedientesReservados
- obtenerExpsRelacionInteresado
- obtenerInteresadoDocumento
- obtenerInteresadoExpediente
- obtenerInteresadoExpedienteCC
- obtenerInteresadoExpedienteDadoDeBaja
- obtenerMensajesUsuario
- obtenerMensajesUsuario

2.4.1.2.36 TrExpedienteCaducado

trewa.bd.trapi.trapiui.tpo.TrExpedienteCaducado

Clase que representa la información referente a un expediente caducado.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ EXPEDIENTE	TrExpediente	Expediente
→ CADUCIDADEX P	TrCaducidadExpediente	Caducidad del expediente
→ FASE	TrFase	Fase que ha superado el tiempo máximo de estancia

Campos definidos para filtrados y ordenación



<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFEXP	NUMÉRICO	Identificador del expediente
→ CAMPO_REFCADEXP	NUMÉRICO	Identificador de la caducidad
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase que ha caducado
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema al que pertenece el expediente
→ CAMPO_FECHACADUCIDAD	DATE	Fecha de caducidad del expediente.
→ CAMPO_NUMEXP	CADENA(30)	Número del expediente
→ CAMPO_TITULOEXP	CADENA(250)	Título del expediente
→ CAMPO_OBSERVACIONESEX	CADENA(250)	Observaciones del expediente
→ CAMPO_REFORGENVP	NUMÉRICO	Identificador del organismo al que pertenece el expediente
→ CAMPO_REFORGENVEX	NUMÉRICO	Identificado del organismo que envía el expediente
→ CAMPO_PROPIETARIOEXP	CADENA(10)	Usuario propietario del expediente
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador del procedimiento al que pertenece el expediente
→ CAMPO_ABREVIATURAPROC	CADENA	Abreviatura del procedimiento al que pertenece el expediente
→ CAMPO_DESCRIPCIONPROC	CADENA	Descripción de la definición del procedimiento
→ CAMPO_DESCRIPCIONAMPPROC	CADENA	Descripción ampliada de la definición del procedimiento
→ CAMPO_CATEGORIAPROC	CADENA(1)	Indica si es familia, subfamilia o procedimiento: → "F" – Familia → "S" – Subfamilia → "P" – Procedimiento
→ CAMPO_COMENTARIOSPROC	CADENA(1)	Indica si es familia, subfamilia o procedimiento: → "F" – Familia → "S" – Subfamilia → "P" – Procedimiento
CAMPO_REFORGANISMOPROC	NUMÉRICO	Organismo al que pertenece
CAMPO_REFORGCMPPROC	NUMÉRICO	Organismo competente
CAMPO_REFORGRESPPROC	NUMÉRICO	Organismo que resuelve
CAMPO_REFORGTRAMPROC	NUMÉRICO	Organismo que tramita
CAMPO_CODWANDAPROC	CADENA	Valor en w@ndA del procedimiento
CAMPO_VIGENTEPROC	CADENA	Indica si el procedimiento está vigente. → "S" – Sí → "N" – No
CAMPO_INFORMARPROC	CADENA	Indica si se tendrá que informar al bus de este procedimiento y de los expedientes que se tramiten. Sus valores pueden ser: → "S" – Sí → "N" – No

CAMPO_REFSUBFAMILI APROC	NUMÉRICO	Identificador de la subfamilia
CAMPO_REFFAMILIARP ROC	NUMÉRICO	Identificador de la familia
CAMPO_OTROSDATOSP ROC	CADENA	Indica si el procedimiento tiene otros datos. Sus valores pueden ser: → “ S ” – Sí → “ N ” – No
CAMPO_USUARIOBLQP ROC	CADENA	Indica el código del usuario que bloquea el procedimiento

Métodos para aplicación de los filtros

→ obtenerExpedientesCaducados

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFEXP	NUMÉRICO	Identificador del expediente
→ CAMPO_NUMEXP	CADENA(30)	Número del expediente
→ CAMPO_TITULOEXP	CADENA(250)	Título del expediente
→ CAMPO_FECHAALTA	DATE	Fecha de alta del expediente
→ CAMPO_OBSERVACIONES	CADENA(250)	Observaciones del expediente
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo al que pertenece el expediente
→ CAMPO_CODORGANISMO	CADENA(16)	Código del organismo al que pertenece el expediente
→ CAMPO_NOMBREORGANISMO	CADENA(32)	Nombre del organismo al que pertenece el expediente
→ CAMPO_DESCORGANISMO	CADENA(200)	Descripción del organismo al que pertenece el expediente
→ CAMPO_CODWANDAORG	CADENA(15)	Código w@ndA del organismo al que pertenece el expediente
→ CAMPO_ORDENPRELACIONOR G	CADENA(15)	Orden de relación del organismo al que pertenece el expediente
→ CAMPO_REFORGENVIA	NUMÉRICO	Identificador del organismo que envía el expediente
→ CAMPO_CODORGENVIA	CADENA(16)	Código del organismo que envía el expediente
→ CAMPO_NOMBREORGENVIA	CADENA(32)	Nombre del organismo que envía el expediente
→ CAMPO_DESCORGENVIA	CADENA(200)	Descripción del organismo que envía el expediente
→ CAMPO_CODWANDAORGENVIA	CADENA(15)	Código w@ndA del organismo que envía el expediente
→ CAMPO_ORDENPRELACIONOR GENVIA	CADENA(15)	Orden de relación del organismo que envía el expediente
→ CAMPO_URLWANDA	CADENA(255)	Url del expediente en w@ndA
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→ CAMPO_DESCDEFPROC	CADENA(50)	Descripción de la definición del procedimiento
→ CAMPO_REFTIPOEXP	NUMÉRICO	Identificador del tipo de expediente
→ CAMPO_DESCTIPOEXP	CADENA(50)	Descripción del tipo de expediente
→ CAMPO_REFSTMADEFPROC	NUMÉRICO	Identificador del sistema de la definición del procedimiento
→ CAMPO_CODSTMADEFPROC	CADENA(10)	Código o nombre del sistema de la definición del procedimiento
→ CAMPO_OTROSDATOS	CADENA(1)	Indica si el expediente tiene otros datos → “ S ” – Sí

→	CAMPO_FECHAARCHIVO	DATE	→ "N" – No Fecha de archivo del expediente
→	CAMPO_REFCOMPONENTE	NUMÉRICO	Identificador del componente dónde está archivado el expediente
→	CAMPO_INFORMADO	CADENA(1)	Indica si se ha informado al bus del expediente → "S" – Sí → "N" – No
→	CAMPO_ABREVTIPOEXP	CADENA(10)	Abreviatura del tipo de expediente
→	CAMPO_ABREVEDEFPROC	CADENA(10)	Abreviatura de la definición del procedimiento
→	CAMPO_PROPIETARIO	CADENA(10)	Usuario propietario del expediente
→	CAMPO_CREADO	CADENA	Usuario que crea el registro
→	CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→	CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→	CAMPO_FECHAMODIFICA	DATE	Fecha de modificación del registro

Métodos para aplicación de los filtros

→ obtenerExpedientes

Campos definidos para las exclusiones

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ EXCL_SISTEMA	Excluye los datos de la entidad sistema de la consulta
→ EXCL_ORGANISMO	Excluye los datos de la entidad organismo del expediente de la consulta
→ EXCL_ORGENVIA	Excluye los datos de la entidad organismo envía del expediente de la consulta.

Métodos para aplicación de exclusiones

→ obtenerExpedientes

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.1.2.37 TrExpedienteSimple

trewa.bd.trapi.trapiui.tpo.TrExpedienteSimple

Clase que representa la información referente a un expediente.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	REFEXP	TpoPK	Identificador del expediente
→	REFSTMA	TpoPK	Identificador de sistema
→	TIPOEXP	TrTipoExoediente	Tipo de expediente
→	DEFPROC	TrDefProcedimiento	Definición del procedimiento
→	FECHAALTA	Timestamp	Fecha de alta
→	NUMEXP	String	Número del expediente en la aplicación "cliente"
→	TITULOEXP	String	Título del expediente en la aplicación "cliente"
→	OBSERVACIONES	String	Observaciones
→	REFORGANISMO	TpoPK	Identificador del organismo al que pertenece
→	REFORGENVIA	TpoPK	Identificador de organismo que lo ha enviado
→	OTROSDATOS	String	Indica si tiene otros datos → " S " – Sí → " N " – No
→	FECHAARCHIVO	Timestamp	Fecha de archivo del expediente
→	INFORMADO	String	Indica si se ha informado del expediente al bus → " S " – Sí → " N " – No
→	REFCOMPONENTE	TpoPK	Identificador del componente en el que está archivado
→	URLWANDA	String	Url del expediente en w@ndA
→	INI_CIUDADANO	Boolean	Iniciado por el ciudadano

Campos definidos para filtrados y ordenación

	<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_ABREVDEFPR OC	CADENA(10)	Abreviatura de la definición de procedimiento
→	CAMPO_ABREVTIPOEX P	CADENA(10)	Abreviatura del tipo de expediente
→	CAMPO_CODSTMADEF PROC	NUMÉRICO	Código del sistema de la definición de procedimiento
→	CAMPO_DESCDEFPRO C	CADENA(50)	Descripción del procedimiento
→	CAMPO_DESCTIPOEXP	CADENA(50)	Descripción del tipo expediente
→	CAMPO_FECHAALTA	DATE	Fecha de alta del expediente



→	CAMPO_FECHAARCHIVO	DATE	Fecha de archivo del expediente
→	CAMPO_INFORMADO	CADENA(1)	Indica si se ha informado al bus del expediente Sus valores pueden ser: → "S" – Sí → "N" – No
→	CAMPO_NOMBREORGANISMO	CADENA(10)	Nombre del organismo
→	CAMPO_NOMBREORGANISMO ENVIA	CADENA(10)	Nombre del organismo que envía
→	CAMPO_NUMEXP	CADENA(10)	Número de expediente
→	CAMPO_OBSERVACIONES	CADENA(10)	Observaciones del expediente
→	CAMPO_OTROSDATOS	CADENA(1)	Indica si el expediente tiene otros datos Sus valores pueden ser: → "S" – Sí → "N" – No
→	CAMPO_REFCOMPONENTE	NUMERICO	Identificador del componente donde está registrado el expediente
→	CAMPO_REFDEFPROC	NUMERICO	Identificador de la definición de procedimiento
→	CAMPO_REFEXP	NUMERICO	Referencia del expediente
→	CAMPO_REFORGANISMO	NUMERICO	Referencia del organismo
→	CAMPO_REFORGENVIA	NUMERICO	Referencia del organismo que envía
→	CAMPO_REFSTMADEFPROC	NUMERICO	Referencia del sistema de la definición de procedimiento
→	CAMPO_REFTIPOEXP	NUMERICO	Referencia del tipo expediente
→	CAMPO_TITULOEXP	NUMERICO	Título del expediente
→	CAMPO_URLWANDA	CADENA	URL w@nda
→	CAMPO_INI_CIUDADANO	Char(1)	Iniciado por el ciudadano

2.4.1.2.38 TrExplorador

trewa.bd.trapi.trapiui.tpo.TrExplorador

Clase que representa la información asociada al estado en el que se encuentra un expediente en fase.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ EXPEDIENTE	TrExpediente	Expediente
→ FASE	TrFase	Fase en la que se encuentra el expediente en fase
→ PENDIENTES	String	Indica si el expediente se encuentra pendiente de alguna tarea o cambio de fase → 'S' – Sí → 'N' – No
→ CADUCADOS	String	Indica si el expediente se encuentra caducado, ya sea por algún plazo definido en el procedimiento o porque ha vencido el plazo de estancia en una fase → 'S' – Sí → 'N' – No
→ RESERVADOS	String	Indica si el expediente se encuentra reservado por el usuario, ya sea una fase del expediente o el expediente completo → 'S' – Sí → 'N' – No
→ ASIGNADO	String	Indica si el usuario está asignado al expediente. → 'S' – Sí → 'N' – No
→ REFDEFPROCEXPFASE	TpoPK	Identificador de la definición del procedimiento en el que se da la fase en la que se encuentra el expediente
→ FECHACADUCIDAD	Timestamp	Fecha de caducidad del expediente
→ USUARIOBLQ	String	Código del usuario que tiene reservado o bloqueado el expediente
→ PARTICIPA	String	Indica si el usuario ha participado en el expediente, es decir ha realizado alguna transición, documento o tarea. → 'S' – Sí → 'N' – No
→ USUARIOASOCIA	String	Código del usuario que dio de alta el expediente o cambió el procedimiento y tipo de expediente del expediente
→ FECHAASOCIA	Timestamp	Fecha de asociación del expediente al procedimiento.
→ OBSERVACIONESASOCIACION	String	Observaciones anotadas en la "historia de evoluciones" por las que ha pasado el expediente.
→ TRANSPERMITIDAS	List TrTransionExplorador	Listado de transiciones permitidas.
→ DOCPERMITIDOS	List TrDocumentoExplora	Listado de documentos permitidos pendientes

dor

- **BLOQPERMITIDOS** List
TrBloquesExplorador Listado de bloques permitidos pendientes.

- **INTERESADOS** List
TrInteresadoExpediente Listado de interesados del expediente.

- **ASIGNADOS** List
TrUsuarioAsignadoExplorador Listado de usuarios asignados al expediente.

- **ABIERTAEVENTO** String Indica si se ha llegado a la fase a través de un evento.
→ **'S'** – Sí
→ **'N'** – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
→ CAMPO_CODSTMA	CADENA(10)	Código del sistema
→ CAMPO_DESCSTMA	CADENA(128)	Descripción del sistema
→ CAMPO_REFTIPOEXP	NUMÉRICO	Identificador del tipo de expediente
→ CAMPO_ABREVTIPOEXP	CADENA(10)	Abreviatura del tipo de expediente
→ CAMPO_DESCTIPOEXP	CADENA(50)	Descripción del tipo de expediente
→ CAMPO_VIGENTETIPOEXP	CADENA(1)	Indica si el tipo de expediente está vigente. Sus valores pueden ser: - "S" – Sí - "N" – No
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→ CAMPO_ABREVDEFPROC	CADENA(10)	Abreviatura de la definición del procedimiento
→ CAMPO_DESCDEFPROC	CADENA(50)	Descripción de la definición del procedimiento
→ CAMPO_VIGENTEDEFPROC	CADENA(1)	Indica si la definición del procedimiento está vigente. Sus valores pueden ser: - "S" – Sí - "N" – No
→ CAMPO_REFEXPEDIENTE	NUMÉRICO	Identificador del expediente
→ CAMPO_NUMEXP	CADENA(30)	Número del expediente
→ CAMPO_TITULOEXP	CADENA(250)	Título del expediente
→ CAMPO_FECHAALTA	DATE	Fecha de alta del expediente
→ CAMPO_OBSERVACIONESEXP	CADENA(250)	Observaciones del expediente
→ CAMPO_REFORGANISMOEXP	NUMÉRICO	Identificador del organismo al que pertenece el expediente



→	CAMPO_REFORGANISMOEN VIAEXP	NUMÉRICO	Identificador del organismo que envía el expediente
→	CAMPO_FECHACADUCIDAD	DATE	Fecha de caducidad del expediente
→	CAMPO_REFFASE	NUMÉRICO	Identificador de la fase
→	CAMPO_NOMBREFASE	CADENA(50)	Nombre de la fase
→	CAMPO_DESCFASE	CADENA(250)	Descripción de la fase
→	CAMPO_TEXTOAUXILIARFA SE	CADENA(500)	Texto auxiliar de la fase
→	CAMPO_ORDENFASE	NUMÉRICO	Orden de la fase
→	CAMPO_REFDEFPROCREUTFA SE	NUMÉRICO	Identificador del procedimiento reutilizables que representa la fase
→	CAMPO_REFMETAFASE	NUMÉRICO	Identificador de la metafase
→	CAMPO_NOMBREMETAFASE	CADENA(50)	Nombre de la metafase
→	CAMPO_PENDIENTES	CADENA(1)	Indica si el usuario está pendiente para el usuario establecido. Sus valores pueden ser: - "S" – Sí - "N" – No
→	CAMPO_CADUCADOS	CADENA(1)	Indica si el expediente está caducado. Sus valores pueden ser: - "S" – Sí - "N" – No
→	CAMPO_RESERVADOS	CADENA(1)	Indica si el expediente está reservado o bloqueado por el usuario establecido. Sus valores pueden ser: - "S" – Sí - "N" – No
→	CAMPO_ASIGNADO	CADENA(1)	Indica si el usuario está asignado al expediente. Sus valores pueden ser: - "S" – Sí - "N" – No
→	CAMPO_REFDEFPROCEXPFA SE	NUMÉRICO	Identificador de la definición del procedimiento en el que se da la fase en la que se encuentra el expediente
→	CAMPO_USUARIOBLQ	CADENA(10)	Código del usuario que tiene reservado o bloqueado el expediente
→	CAMPO_PARTICIPA	CADENA(1)	Indica si el usuario participa en el expediente, es decir ha realizado alguna transición, documento o tarea.
→	CAMPO_USUARIOASOCIA	CADENA(10)	Código del usuario que asocia el expediente al procedimiento, es decir el que dio de alta el expediente o lo cambió de procedimiento.
→	CAMPO_FECHAASOCIA	DATE	Fecha en la que asocia el expediente al procedimiento, es decir el que dio de alta el expediente o se cambió de procedimiento.
→	CAMPO_OBSERVACIONESASOCIA	CADENA(250)	Observaciones de la asociación del expediente al procedimiento.
→	CAMPO_PROPIETARIOEXP	CADENA(10)	Usuario propietario del expediente
→	CAMPO_ABIERTAEVENTO	CADENA(1)	Indica si se ha llegado a al fase a través de un evento. Sus valores pueden ser: - "S" – Sí - "N" – No
→	CAMPO_FECHAARCHIVOEXP	DATE	Fecha de archivo del expediente
→	CAMPO_IDENI	CADENA	Identificador ENI que pertenece al expediente

- **CAMPO_INFORMADOEXP** CADENA Indica si el expediente ha sido informado al bus
- "S" – Sí
- **CAMPO_INFPENDIENTE** CADENA Indica si el tipo de expediente está vigente. Sus valores pueden ser:
- "N" – No
- **CAMPO_REFCOMPARCHIVAE XP** NUMÉRICO Identificador del componente en el que está archivado el expediente
- "S" – Sí
- **CAMPO_URLWANDAEXP** CADENA Url del expediente en w@ndA
- "N" – No

Métodos para aplicación de los filtros

→ explorarExpedientes

Campos definidos para las exclusiones

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ EXCL_CADUCIDADES	Excluye los datos de la entidad caducidades de la consulta
→ EXCL_PENDIENTE	Excluye los datos de expedientes pendientes de la consulta
→ EXCL_PARTICIPA	Excluye los datos de usuario participante en el expediente de la consulta
→ EXCL_ASIGNADO	Excluye los datos de usuario asignado al expediente de la consulta
→ EXCL_SISTEMA	Excluye los datos de la entidad sistema de la consulta
→ EXCL_TIPOEXPEDIENTE	Excluye los datos de la entidad tipo de expediente de la consulta
→ EXCL_DEFPROCEDIMIENTO	Excluye los datos de la entidad de procedimiento de la consulta
→ EXCL_METAFASE	Excluye los datos de la entidad metafase de la consulta
→ EXCL_INTE	Excluye los datos de los interesados del expediente
→ EXCL_ASIGNADOS	Excluye los datos de los usuarios asignados al expediente
→ EXCL_TRANSER	Excluye los datos de las transiciones permitidas en la fase
→ EXCL_BLOPPER	Excluye los datos de los bloques permitidos en la fase
→ EXCL_DOCPER	Excluye los datos de los documentos permitidos en la fase

Métodos para aplicación de exclusiones

→ explorarExpedientes

Campos definidos para aplicar filtros por datos propios del interesado

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ FILTRO_NUMIDENT_INTE	Número de identificación del interesado
→ FILTRO_APELLIDO1_INTE	Primer apellido del interesado
→ FILTRO_APELLIDO2_INTE	Segundo apellido del interesado
→ FILTRO_NOMBRE_INTE	Nombre del interesado
→ FILTRO_CIWA_INTE	Código w@ndA del interesado
→ FILTRO_COMENTARIOS_INTE	Comentarios del interesado

- **FILTRO_ANAGRAMAFISCAL_INTE** Anagrama fiscal del interesado
- **FILTRO_OBSERVACIONES_INTE** Observaciones del interesado
- **FILTRO_ABREVIATURA_RAZON** Abreviatura de la razón de interés del interesado
- **FILTRO_CODWANDA_RAZON** Código w@ndA de la razón de interés del interesado

Métodos para aplicación de exclusiones → explorarExpedientes

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.1.2.39 TrFamiliaSubfamilia

trewa.bd.trapi.trapiui.tpo.TrFamiliaSubfamilia

Clase que representa la información de una familia o subfamilia de un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento
→ ABREVIATURA	String	Abreviatura de la definición del procedimiento.
→ DESCRIPCION	String	Descripción de la definición del procedimiento.
→ CATEGORIA	String	Indica la categoría del procedimiento → ' F ' – Familia → ' S ' – Subfamilia → ' P ' – Procedimiento
→ DESCRIPCIONA MP	String	Descripción ampliado
→ CODWANDA	String	Valor en w@ndA del procedimiento

Métodos para uso de la clase → obtenerFamiliaSubfamilia

2.4.1.2.40 TrFase

trewa.bd.trapi.trapiui.tpo.TrFase

Clase que representa la información asociada a una fase.

El modelo de referencia [w@nda](#) define una fase como “un conjunto homogéneo de tareas desde el punto de vista del tramitador, cuya sucesión en el tiempo componen un determinado procedimiento una vez ha sido modelado y constituyen la unidad elemental de tramitación dentro del mismo”.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	AUDITORIA	TrAuditoria	Datos de auditoria
→	REFFASE	TpoPK	Identificador de la fase
→	NOMBRE	String	Nombre de la fase
→	DESCRIPCION	String	Descripción de la fase
→	METAFASE	TrMetafase	Metafase
→	DEFPROC	TrDefProcedimiento	Definición del procedimiento
→	STMA	TrSistema	Sistema
→	INFORMARBUS	String	Indica si se informa al bus de la entrada en la fase → ' S ' – Sí → ' N ' – No
→	TEXTOAUXILIAR	String	Texto auxiliar de la fase
→	ORDEN	Integer	Orden de la fase
→	TELEMATICA	String	Indica si es de origen telemático → ' S ' – Sí → ' N ' – No
→	INFORMACION PUBLICA	String	Indica si es de información pública. → ' S ' – Sí → ' N ' – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la fase
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la fase
→ CAMPO_METAFASE	CADENA(50)	Nombre de la metafase
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
→ CAMPO_TEXTOAUXILIAR	CADENA(500)	Texto auxiliar para la fase
→ CAMPO_ORDENFASE	NUMÉRICO	Orden de la fase
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento si la fase representa un procedimiento reutilizable
→ CAMPO_DESCDEFPROC	CADENA(50)	Descripción de la definición del procedimiento si la fase representa un procedimiento reutilizable
→ CAMPO_REFMETAFASE	NUMÉRICO	Identificador de la metafase
→ CAMPO_DESCMETAFASE	CADENA(250)	Descripción de la metafase
→ CAMPO_ORDENMETAFAS	NUMÉRICO	Orden de la metafase
E		
→ CAMPO_INFORMARBUSF ASE	CADENA(1)	Indica si se informa al bus de la entrada en la fase → ' S ' – Sí → ' N ' – No
→ CAMPO_CAMPO_INFORM ARBUSMETAFASE	CADENA(1)	Indica si se informa al bus de la entrada en la metafase → ' S ' – Sí → ' N ' – No
→ CAMPO_TELEMATICA	CADENA(1)	Indica si es de origen telemático → ' S ' – Sí → ' N ' – No
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICA	DATE	Fecha de modificación del registro
→ CAMPO_INFORMACIONP UBLICA	CADENA(1)	Indica si la fase es de información pública. Sus valores pueden ser: → ' S ' – Sí → ' N ' – No

Métodos para aplicación de los filtros

→ obtenerDatosFase, obtenerFasesFinTransicion

2.4.1.2.41 TrFaseActualExpediente

trewa.bd.trapi.trapiui.tpo.TrFaseActualExpediente

Clase que representa una situación en la que se encuentra un expediente.

El modelo de referencia [w@nda](#) define una fase actual como “la fase activa en estos momentos en el tramitador de procedimientos para el expediente de todas las posibles definidas en el procedimiento”.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ FASE	TrFase	Fase
→ FECHAENTRADA	Timestamp	Fecha en la que se entró en la fase
→ FECHALIMITE	Timestamp	Fecha de salida de la fase
→ USUARIO	String	Usuario que lo dejó en la fase
→ USUARIOBLQ	String	Usuario que lo tiene bloqueado en la fase
→ OBSERVACIONES	String	Observaciones del usuario que lo dejó en la fase
→ REFEXPXFAS	TpoPK	Identificador de la situación actual de la evolución total del expediente. Es el id del paso concreto de la tramitación.
→ REFFASEPADRE	TpoPK	Para el caso en que la situación pertenezca a un módulo reutilizable, indica la fase que representa el módulo reutilizable
→ REFDEFPROC	TpoPK	Para el caso que la fase pertenezca a un módulo reutilizable, indica cuál es dicho módulo.
→ REFEXPXFASPADRE	TpoPK	Identificador del expediente en fase padre
→ REFTRANSICION	TpoPK	Identificador de la transición
→ DESCTRANSICION	String	Descripción de la transición
→ REFTIPOACTO	TpoPK	Identificador del tipo de acto
→ TIPOTRANS	String	Tipo de la transición. → “ N ” – Transición normal → “ D ” – Transición de división → “ U ” – Transición de unión → “ ES ” – Evento que provoca salida → “ EN ” – Evento que no provoca la salida
→ NOMBREUSU	String	Nombre y apellidos del usuario
→ NOMBREUSUBLQ	String	Nombre y apellidos del usuario que bloquea
→ ABIERTAEVENTO	String	Indica si la fase viene propagada de un evento → “ S ” – Sí → “ N ” – No
→ INFORMADABUS	String	Indica si se ha informado al bus → “ S ” – Sí → “ N ” – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase o situación
→ CAMPO_NOMBFASE	CADENA(50)	Nombre de la fase
→ CAMPO_REFMETAFASE	NUMÉRICO	Identificador de la metafase
→ CAMPO_NOMBMETAFASE	CADENA(50)	Nombre de la metafase
→ CAMPO_FECHAENTRADA	DATE	Fecha en la que se entró en la fase
→ CAMPO_FECHALIMITE	DATE	Fecha de salida de la fase

→	CAMPO_USUARIO	CADENA(10)	Usuario que lo dejó en la fase
→	CAMPO_USUARIOBLQ	CADENA(10)	Usuario que lo tiene bloqueado en la fase
→	CAMPO_OBSERVACIONES	CADENA(250)	Observaciones del usuario que lo dejó en la fase
→	CAMPO_REFEXPXFAS	NUMÉRICO	Identificador de la situación actual en la evolución total del expediente. Es el identificador del paso concreto en la tramitación.
→	CAMPO_REFFASEPADRE	NUMÉRICO	Identificador de la fase padre
→	CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→	CAMPO_REFEXPXFASPADRE	NUMÉRICO	Identificador del expediente en fase padre
→	CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición
→	CAMPO_DESCTRANSICION	CADENA(50)	Descripción de la transición
→	CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto
→	CAMPO_TIPOTRANS	CADENA(2)	Tipo de la transición. → “ N ” – Transición normal → “ D ” – Transición de división → “ U ” – Transición de unión → “ ES ” – Evento que provoca salida → “ EN ” – Evento que no provoca la salida
→	CAMPO_DESCFASE	CADENA(250)	Descripción de la fase
→	CAMPO_NOMBREUSUARIO	CADENA(98)	Nombre y apellidos del usuario
→	CAMPO_NOMBREUSUARIOBLQ	CADENA(98)	Nombre y apellidos del usuario que bloquea
→	CAMPO_ABIERTAEVENTO	CADENA(1)	Indica si la fase viene propagada de un evento → “ S ” – Sí → “ N ” – No
→	CAMPO_TEXTOAUXILIAR	CADENA(500)	Texto auxiliar de la fase
→	CAMPO_INFORMADABUS	CADENA(1)	Indica si se informa al bus

Métodos para aplicación de los filtros

- obtenerFaseActualExpediente
- modificarDatosFaseActual

2.4.1.2.42 TrFaseExpediente

trewa.bd.trapi.trapiui.tpo.TrFaseExpediente

Clase que representa la información referente a situaciones por las que pasa un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
------------------------	--------------------	---------------------------

→	REFEXPEDIENTE	TpoPK	Identificador del expediente
→	NUMEXP	String	Número del expediente
→	TITULOEXP	String	Título del expediente
→	OBSERVACIONESEX P	String	Observaciones del expediente
→	REFORGENVP	TpoPK	Identificador del organismo al que pertenece el expediente
→	REFORGENVEXP	TpoPK	Identificador del organismo que envía el expediente
→	REFEXPXNAS	TpoPK	Identificador del paso en la evolución
→	FASE	TrFase	Fase o situación
→	FECHAENTRADA	Timestamp	Fecha en la que se entró en la fase
→	FECHASALIDA	Timestamp	Fecha en la que se salió de la fase
→	FECHALIMITE	Timestamp	Fecha límite de salida de la fase
→	USUARIO	String	Usuario que lo dejó en la fase
→	USUARIOBLQ	String	Usuario que lo tiene bloqueado en la fase
→	OBSERVACIONES	String	Observaciones del usuario que lo dejó en la fase
→	REFEXPXNASPADR E	TpoPK	Para los pasos pertenecientes a módulos reutilizables, indica el paso que dejó al expediente en la fase que representa a dicho módulo.
→	TRANSICION	TrTransicion	Transición
→	INFORMADABUS	String	Indica si se informa al bus de la entrada en la fase → “S” – Sí → “N” – No
→	INFORMADAFINBU S	String	Indica si se informa al bus de la salida de la fase → “S” – Sí → “N” – No
→	REFFASEPADRE	TpoPk	Identificador de la fase padre
→	NOMBREUSU	String	Nombre y apellidos del usuario
→	REFDEFPROC	TpoPK	Identificador de la definición del procedimiento
→	ABIERTAEVENTO	String	Indica si la fase viene propagada de un evento → “S” – Sí → “N” – No
→	PROPIETARIOEXP	String	Usuario propietario del expediente

Campos definidos para filtrados y ordenación

	<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_REFEXPEDIENTE	NUMÉRICO	Identificador del expediente
→	CAMPO_NUMEXP	CADENA(30)	Número del expediente
→	CAMPO_TITULOEXP	CADENA(250)	Título del expediente
→	CAMPO_OBSERVACIONESEX SEXP	CADENA(250)	Observaciones del expediente
→	CAMPO_REFORGENVP	NUMÉRICO	Identificador del organismo al que pertenece el expediente
→	CAMPO_REFORGENVEXP P	NUMÉRICO	Identificador del organismo que envía el expediente
→	CAMPO_ESTADOARCEXP P	CADENA(1)	Estado de la petición de archivado Sus valores pueden ser: → “p” – Pendiente



			→ “ A ” – Archivado
			→ “ R ” – Rechazado
→	CAMPO_FECHAARCHEX	DATE	Fecha de archivo del expediente
	P		
→	CAMPO_REFEXPXFAS	NUMÉRICO	Identificador del paso en la evolución
→	CAMPO_REFFASE	NUMÉRICO	Identificador de la fase o situación
→	CAMPO_FASE	CADENA(50)	Nombre de la fase
→	CAMPO_REFMETAFASE	NUMÉRICO	Identificador de la metafase
→	CAMPO_FECHAENTRAD	DATE	Fecha en la que se entró en la fase
	A		
→	CAMPO_FECHASALIDA	DATE	Fecha en la que se salió de la fase
→	CAMPO_FECHALIMITE	DATE	Fecha límite de salida de la fase
→	CAMPO_USUARIO	CADENA(10)	Usuario que lo dejó en la fase
→	CAMPO_USUARIOBLQ	CADENA(10)	Usuario que lo tiene bloqueado en la fase
→	CAMPO_OBSERVACIONE	CADENA(250)	Observaciones del usuario que lo dejó en la fase
	S		
→	CAMPO_REFTRANSICIO	NUMÉRICO	Identificador de la transición
	N		
→	CAMPO_DESCTRANSICI	CADENA(50)	Descripción de la transición
	ON		
→	CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto administrativo que representa la transición
→	CAMPO_REFEXPXFASPA	NUMÉRICO	Para los pasos pertenecientes a módulos reutilizables, indica el paso que dejó al expediente en la fase que representa a dicho módulo.
	DRE		
→	CAMPO_TIPOTRANS	CADENA(2)	Indica el tipo de la transición
			→ “ N ” – Transición normal
			→ “ D ” – Transición de división
			→ “ U ” – Transición de unión
			→ “ ES ” – Evento que provoca salida
			→ “ EN ” – Evento que no provoca la salida
→	CAMPO_REFFASEPADRE	NUMÉRICO	Identificador de la fase padre
→	CAMPO_DESCFASE	CADENA(250)	Descripción de la fase
→	CAMPO_NOMBREUSU	CADENA(98)	Nombre y apellidos del usuario
→	CAMPO_NOMBREUSUBL	CADENA(98)	Nombre y apellidos del usuario que bloquea
	Q		
→	CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→	CAMPO_ABIERTAEVENT	CADENA(1)	Indica si la fase viene propagada de un evento
	O		→ “ S ” – Sí
			→ “ N ” – No
→	CAMPO_INFORMADABU	CADENA(1)	Indica si se informa al bus de la entrada en la fase
	S		→ “ S ” – Sí
			→ “ N ” – No
→	CAMPO_INFORMADAFIN	CADENA(1)	Indica si se informa al bus de la salida de la fase
	BUS		→ “ S ” – Sí
			→ “ N ” – No
→	CAMPO_TEXTOAUXILIA	CADENA(500)	Texto auxiliar de la fase
	R		
→	CAMPO_PROPIETARIOE	CADENA(10)	Usuario propietario del expediente
	XP		

Métodos para aplicación de los filtros

→ obtenerFasesExpediente

2.4.1.2.43 TrFirmaDocumentoExpediente

trewa.bd.trapi.trapiui.tpo.TrFirmaDocumentoExpediente

Clase que representa la información referente a los datos de la firma de un documento.

Atributos accesibles mediante métodos get/set

	Atributo	Tipo	Descripción
→	FECHA	Timestamp	Fecha de firma
→	FIRMANTEPPA L	TrFirmante	Firmante principal
→	USUARIO	String	Clave del usuario
→	USUARIODIG	String	Usuario al que pertenece la unidad orgánica
→	REFINTERESAD O	TpoPK	Identificador del interesado que realiza la firma
→	FIRMANTEDEL EG	TrFirmante	Firmante delegado
→	FIRMANTESUS T	TrFirmante	Firmante sustituto
→	EDITABLE	String	Indica si es editable el pie de firma → 'S' – El pie de firma es editable → 'N' – El pie de firma no es editable
→	TXTPIE	String	Texto que va antes de la rúbrica de la firma en el pie de firma
→	TXTFDO	String	Texto que va después de la rúbrica de la firma en el pie de firma
→	USUARIODIGA NT	String	Código del usuario digital de la firma anterior
→	REFUNIORGAN T	TpoPK	Identificador de la unidad orgánica de la firma anterior
→	PTOTRABANT	String	Código del puesto de trabajo de la firma anterior
→	HASH	String	Código de petición para port@firmas
→	FECHAFIRMADI G	Timestamp	Fecha de la firma digital
→	CODTRANSACC ION	String	Código de transacción de firma
→	PKCS7	byte[]	Recibo de la firma digital
→	ORDEN	Integer	Orden de la firma
→	FIRMA	TrFirma	Firma del usuario digital
→	PUESTO	String	Nombre del puesto de trabajo del usuario digital
→	ORGANISMO	String	Nombre del organismo del usuario digital
→	NOMBREUSUDI	String	Nombre y apellidos del usuario digital.

GI		
→	REFDOCEXP	TpoPK Identificador del documento
→	TIPOFIRMA	String Indica el tipo de firma. → La firma clásica "N". → Firma de interesado "I". → Firma dinámica "D". → Expedientes ENI "E".
→	TIPOFIRMAENI	String Indica el tipo de firma ENI. → TF01- CSV → TF02 - XAdES internally detached signature. → TF03 - XAdES enveloped signature. → TF04 - CAdES detached/explicit signature. → TF05 - CAdES attached/implicit signature. → TF06 - PAdES.
→	CODIGOCSV	String Código seguro de verificación
→	REFDOCEXPAN	TpoPK Identificador del documento expediente de la firma anterior
T		
→	REFFIRMAEOE	TpoPK Referencia de la firma documento expediente
XP		
→	REGULACIONC	String Regulación del CSV
SV		
→	TXTAUX	String Texto auxiliar para validación de servidor de firmas
→	V_TIPO	String Tipo de firma relativa al firmante

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_FECHA	DATE Fecha de firma
→	CAMPO_PTOTRABAJO	CADENA(10) Código que identifica el puesto de trabajo
→	CAMPO_UNIORGANICA	NUMÉRICO Código que identifica la unidad orgánica
→	CAMPO_USUARIO	CADENA(10) Clave de usuario
→	CAMPO_USUARIODIG	CADENA(10) Usuario al que pertenece la unidad orgánica
→	CAMPO_INTERESADO	CADENA(10) Interesado que realiza la firma
→	CAMPO_PTOTRABDEL	CADENA(10) Código que identifica el puesto de trabajo del delegado
EG		
→	CAMPO_UNIORGDELE	NUMÉRICO Código que identifica la unidad orgánica del delegado
G		
→	CAMPO_PTOTRABSUS	CADENA(10) Código que identifica el puesto de trabajo del sustituto
T		
→	CAMPO_UNIORGST	NUMÉRICO Código que identifica la unidad orgánica del sustituto
→	CAMPO_EDITABLE	CADENA(1) Indica si es editable el pie de firma → ' S ' – El pie de firma es editable → ' N ' – El pie de firma no es editable
→	CAMPO_TXTPIE	CADENA(1000) Texto que va antes de la rúbrica de la firma en el pie de firma
→	CAMPO_TXTFDO	CADENA(250) Texto que va después de la rúbrica de la firma en el pie de firma
→	CAMPO_USUARIODIGA	CADENA(10) Código del usuario digital de la firma anterior
NT		
→	CAMPO_REFUNIORGA	NUMÉRICO Identificador de la unidad orgánica de la firma anterior
NT		

→	CAMPO_PTOTRABANT	CADENA(10)	Código del puesto de trabajo de la firma anterior
→	CAMPO_HASH	CADENA(20)	Código de petición para port@firmas
→	CAMPO_FECHAFIRMA DIG	DATE	Fecha de la firma digital
→	CAMPO_CODTRANSAC	CADENA(250)	Código de transacción de firma
→	CAMPO_ORDEN	NUMÉRICO	Orden de la firma
→	CAMPO_PUESTO	CADENA(32)	Puesto de trabajo del usuario que firma digitalmente
→	CAMPO_ORGANISMO	CADENA(32)	Organismo al que pertenece el usuario que firma digitalmente
→	CAMPO_NOMBREUSUD IGI	CADENA(100)	Nombre y apellidos del usuario digital.
→	CAMPO_FORMATO	CADENA(128)	Formato de la firma digital
→	CAMPO_NOMBREARCH IVO	CADENA(64)	Nombre del archivo de la firma digital.
→	CAMPO_REFDOCEXP	NUMÉRICO	Identificador del documento
→	CAMPO_TIPOFIRMA	CADENA(1)	Indica el tipo de firma. → La firma clásica "N". → Firma de interesado "I". → Firma dinámica "D". → Expedientes ENI "E".
→	CAMPO_VTIPO	CADENA(2)	Tipo de firma relativa al firmante. Pueden ser: → ' F ' – El tipo de firma del firmante es "firma" → ' VB ' – El tipo de firma del firmante es "visto bueno"

→ firmarDinamicamenteDocumento

→ firmarDocumento

→ firmarDocumentoInteresado

Métodos para aplicación de los filtros y uso de la clase

→ incluirFirmaDigitalDocumentoDelegado

→ importarDocumento

→ obtenerDatosFirmasDocumento

→ obtenerFirmasDocumento

→ obtenerUltimaFirma

2.4.1.2.44 TrFirmante

trewa.bd.trapi.trapiui.tpo.TrFirmante

Clase que representa la información referente a un firmante.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	REFFIRMANTE	TpoPK	Identificador del firmante definido
→	PTOTRABAJO	String	Código que identifica el puesto de trabajo
→	UNIORGANICA	TpoPK	Código que identifica a la unidad orgánica
→	TIPO	String	Tipo de firmante definido → ' P ' – Principal → ' D ' – Delegado

→	FECHAINIVIG	Timestamp	→ ' S ' – Sustituto
→	FECHAFINIVIG	Timestamp	Fecha de inicio de vigencia
→	TEXTODISPOSICION	TrTextoDisposicion	Fecha de fin de vigencia
→	REFDELEGSUST	TpoPK	Texto de la disposición
→	PUESTO	String	Identificador del firmante que actúa como delegado o sustituto
→	ORGANISMO	String	Nombre del puesto de trabajo
→	CODIGOUSUARIO	String	Nombre del organismo o unidad orgánica
→	IDENTIFICADORUSUARIO	String	Código del usuario
→	NOMBREUSUARIO	String	Identificador del usuario
→			Nombre del usuario

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFFIRMANTE	NUMÉRICO	Identificador del firmante definido
→ CAMPO_PTOTRABAJO	CADENA(10)	Código que identifica el puesto de trabajo
→ CAMPO_UNIORGANICA	NUMÉRICO(10)	Código que identifica a la unidad orgánica
→ CAMPO_TIPO	CADENA(1)	Tipo de firmante definido → ' P ' – Principal → ' D ' – Delegado → ' S ' – Sustituto
→ CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia
→ CAMPO_FECHAFINIVIG	DATE	Fecha de fin de vigencia
→ CAMPO_TEXTODISPOSICION	NUMÉRICO	Identificador del texto de la disposición
→ CAMPO_REFDELEGSUST	NUMÉRICO	Identificador del firmante que actúa como delegado o sustituto
→ CAMPO_PUESTO	CADENA(32)	Nombre del puesto de trabajo
→ CAMPO_ORGANISMO	CADENA(32)	Nombre del organismo o unidad orgánica
→ CAMPO_DESCTEXTODISP	CADENA(100)	Texto de la disposición
→ CAMPO_TIPODISP	CADENA(1)	Tipo de la disposición → ' D ' – Decreto → ' O ' – Orden → ' T ' – Otros

Métodos para aplicación de los filtros y dónde se utiliza

- firmarDinamicamenteDocumento
- firmarDocumento
- obtenerFirmantesDefinidos
- obtenerFirmantesFirmaDinamica
- obtenerFirmantesTipoDocumento
- obtenerFirmasDocumento
- obtenerUltimaFirma

2.4.1.2.45 TrFirmanteTipoDocumento

trewa.bd.trapi.trapiui.tpo.TrFirmanteTipoDocumento

Clase que representa la información referente a un firmante definido para un tipo de documento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ FIRMANTE	TrFirmante	Firmante definido
→ TIPODOC	TrTipoDocumento	Tipo de documento
→ EDITABLE	String	Indica si el pie de firma es editable
→ ETIQUETA	String	Permite poner una etiqueta en el pie de firma
→ ORDEN	long	Orden de la firma en el documento
→ TIPO	String	Orden de la firma en el documento.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFFIRMANTE	NUMÉRICO	Identificador de la indicación de la ficha del procedimiento.
→ CAMPO_CODPTOTRABAJO	CADENA	Identificador de la definición del procedimiento.
→ CAMPO_PUESTOTRABAJO	CADENA	Identificador del tipo de indicación de la ficha.
→ CAMPO_REFORGANISMO	NUMÉRICO	Descripción de la indicación de la ficha del procedimiento.
→ CAMPO_ORDEN	CADENA	Descripción del tipo de indicación de la ficha del procedimiento.
→ CAMPO_EDITABLE	CADENA	Indica si el tipo de indicación de la ficha del procedimiento está obsoleto "S" – Sí "N" – No
→ CAMPO_ETIQUETA	CADENA	Código w@ndA del tipo de indicación de la ficha del procedimiento.
→ CAMPO_TIPO	CADENA	Tipo de firma. Puede ser: "F" – La firma es de tipo "firma" "VB" – La firma es de tipo "visto bueno"
→ CAMPO_REFTEXTODISP	NUMÉRICO	Identificador del texto de disposición de firma
→ CAMPO_TEXTODISP	CADENA	Texto de la disposición de firma.
→ CAMPO_TIPOTEXTODISP	CADENA	Tipo del texto de la disposición de firma.
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento.
→ CAMPO_ETIQUETATIPODOC	CADENA	Etiqueta del tipo de documento.
→ CAMPO_NOMBRETIPODOC	CADENA	Nombre del tipo de documento.
→ CAMPO_DESCTIPODOC	CADENA	Descripción del tipo de documento.

Métodos para aplicación de los filtros

→ obtenerFirmantesTipoDocumento

2.4.1.2.46 TrIndicacionFicha

trewa.bd.trapi.trapiui.tpo.TrIndicacionFicha

Clase que representa datos variables sobre la ficha descriptiva del procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFINDFICHA	TpoPK	Identificador de la indicación de la ficha del procedimiento.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ TIPOINDICA	TrTipoIndicacion	Tipo de indicación de la ficha.
→ DESCRIPCION	String	Descripción de la indicación de la ficha del procedimiento.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFINDFICHA	NUMÉRICO	Identificador de la indicación de la ficha del procedimiento.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_REFTIPOINDICA	NUMÉRICO	Identificador del tipo de indicación de la ficha.
→ CAMPO_DESCRIPCION	CADENA(300)	Descripción de la indicación de la ficha del procedimiento.
→ CAMPO_ABREVTIPOIND	CADENA(10)	Abreviatura del tipo de indicación de la ficha del procedimiento.
→ CAMPO_DESCTIPOIND	CADENA(100)	Descripción del tipo de indicación de la ficha del procedimiento.
→ CAMPO_OBSOLETOTIPOIND	CADENA(1)	Indica si el tipo de indicación de la ficha del procedimiento está obsoleto "S" – Sí "N" – No
→ CAMPO_CODWANDATIPOIND	CADENA(30)	Código w@ndA del tipo de indicación de la ficha del procedimiento.

Métodos para aplicación de los filtros

→ obtenerIndicacionFicha

2.4.1.2.47 TrInteresado

trewa.bd.trapi.trapiui.tpo.TrInteresado

Clase que representa los datos principales de los interesados que se recogen en el sistema.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFINTERESADO	TpoPK	Identificador del interesado
→ TIPOIDENT	String	Tipo de identificación

→	NUMIDENT	String	Número de identificación.
→	APELLIDO1	String	Primer apellido del interesado.
→	APELLIDO2	String	Segundo apellido del interesado.
→	NOMBRE	String	Nombre del interesado.
→	SEXO	String	Sexo del interesado. → 'F' – Femenino → 'M' – Masculino
→	DIGCONTROL	String	Dígito de control para el número de identificación.
→	CIWA	String	Código de identificación w@ndA.
→	FECHANACIM	Timestamp	Fecha de nacimiento.
→	FECHABAJA	Timestamp	Fecha de baja en el sistema.
→	NUMREGMER	String	Número del registro mercantil.
→	FECHARCA	Timestamp	Fecha de RCA de la entidad.
→	EPIGRAFEIAE	String	Epígrafe IAE.
→	EPIGRAFECNAE	String	Epígrafe CNAE.
→	COMENTARIOS	String	Comentarios.
→	OTROSDATOS	String	Indica si el interesado tiene otros datos → 'S' – Sí → 'N' – No
→	TIPOORGZ	String	Tipo de organización.
→	ANAGRAMAFISCAL	String	Anagrama fiscal del interesado
→	REFDATOCONT	TpoPK	Identificador de los datos de contacto
→	RAZONSOCIAL	String	Razón social de la empresa
→	TIPOIDENTEMPR	String	Tipo identificación de la empresa
→	NUMIDENTEMPR	String	Número identificación de la empresa
→	MOTIVOB AJA	String	Motivo por el cual se da de baja un interesado

Campos definidos para filtrados y ordenación

	<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_REFINTERESADO	NUMÉRICO	Identificador del interesado
→	CAMPO_TIPOIDENT	CADENA(10)	Tipo de identificación
→	CAMPO_NUMIDENT	CADENA(15)	Número de identificación.
→	CAMPO_APELLIDO1	CADENA(50)	Primer apellido del interesado.
→	CAMPO_APELLIDO2	CADENA(50)	Segundo apellido del interesado.
→	CAMPO_NOMBRE	CADENA(100)	Nombre del interesado.
→	CAMPO_SEXO	CADENA(1)	Sexo del interesado. → 'F' – Femenino → 'M' – Masculino
→	CAMPO_DIGCONTROL	CADENA(1)	Dígito de control para el número de identificación.
→	CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA.
→	CAMPO_FECHANACIM	DATE	Fecha de nacimiento.
→	CAMPO_FECHABAJA	DATE	Fecha de baja en el sistema.
→	CAMPO_NUMREGMER	CADENA(30)	Número del registro mercantil.
→	CAMPO_FECHARCA	DATE	Fecha de RCA de la entidad.

- **CAMPO_EPIGRAFEIAE** CADENA(30) Epígrafe IAE.
- **CAMPO_EPIGRAFECNAE** CADENA(30) Epígrafe CNAE.
- **CAMPO_COMENTARIOS** CADENA(100) Comentarios.
- **CAMPO_OTROSDATOS** CADENA(1) Indica si el interesado tiene otros datos
→ '**S**' – Sí
→ '**N**' – No
- **CAMPO_TIPOORGZ** CADENA(1) Tipo de organización.
- **CAMPO_ANAGRAMAFISCAL** CADENA(20) Anagrama fiscal del interesado
- **CAMPO_REFDATOCONT** NUMÉRICO Identificador de los datos de contacto
- **CAMPO_RAZONSOCIAL** CADENA(100) Razón social de la empresa
- **CAMPO_TIPOIDENTEMPR** CADENA(10) Tipo identificación de la empresa
- **CAMPO_NUMIDENTEMPR** CADENA(15) Número identificación de la empresa
- **CAMPO_MOTIVOBaja** CADENA(100) Motivo por el que se da de baja un interesado

- obtenerInteresado
- obtenerInteresadoDocumento
- explorarExpedientes
- obtenerInteresadoExpediente
- obtenerInteresadoExpedienteCC
- obtenerInteresadoExpedienteDadoDeBaja
- obtenerInteresados
- obtenerRelacionesInteresado

Métodos para aplicación de los filtros y dónde se utiliza la clase

2.4.1.2.48 TrInteresadoSimple

trewa.bd.trapi.trapiui.tpo.TrInteresadoSimple

Clase que representa los datos principales de los interesados que se recogen en el sistema, exceptuando los datos de interesado jurídico (empresa), los cuales deberán crearse añadiéndolos como un nuevo interesado y creando la relación entre ambos interesados.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFINTERESADO	TpoPK	Identificador del interesado
→ TIPOIDENT	String	Tipo de identificación
→ NUMIDENT	String	Número de identificación.
→ APELLIDO1	String	Primer apellido del interesado.
→ APELLIDO2	String	Segundo apellido del interesado.
→ NOMBRE	String	Nombre del interesado.
→ SEXO	String	Sexo del interesado. → 'F' – Femenino → 'M' – Masculino
→ DIGCONTROL	String	Dígito de control para el número de identificación.
→ CIWA	String	Código de identificación w@ndA.
→ FECHANACIM	Timestamp	Fecha de nacimiento.
→ FECHABAJA	Timestamp	Fecha de baja en el sistema.
→ NUMREGMER	String	Número del registro mercantil.
→ FECHARCA	Timestamp	Fecha de RCA de la entidad.
→ EPIGRAFEIAE	String	Epígrafe IAE.
→ EPIGRAFCNAE	String	Epígrafe CNAE.
→ COMENTARIOS	String	Comentarios.
→ OTROSDATOS	String	Indica si el interesado tiene otros datos → 'S' – Sí → 'N' – No
→ TIPOORGZ	String	Tipo de organización.
→ ANAGRAMAFISCAL	String	Anagrama fiscal del interesado
→ REFDATOCONT	TpoPK	Identificador de los datos de contacto
→ MOTIVOBaja	String	Motivo de la baja

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFINTERESADO	NUMÉRICO	Identificador del interesado
→ CAMPO_TIPOIDENT	CADENA(10)	Tipo de identificación
→ CAMPO_NUMIDENT	CADENA(15)	Número de identificación.
→ CAMPO_APELLIDO1	CADENA(50)	Primer apellido del interesado.
→ CAMPO_APELLIDO2	CADENA(50)	Segundo apellido del interesado.
→ CAMPO_NOMBRE	CADENA(100)	Nombre del interesado.

→	CAMPO_SEXO	CADENA(1)	Sexo del interesado. → 'F' – Femenino → 'M' – Masculino
→	CAMPO_DIGCONTROL	CADENA(1)	Dígito de control para el número de identificación.
→	CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA.
→	CAMPO_FECHANACIM	DATE	Fecha de nacimiento.
→	CAMPO_FECHABAJA	DATE	Fecha de baja en el sistema.
→	CAMPO_NUMREGMER	CADENA(30)	Número del registro mercantil.
→	CAMPO_FECHARCA	DATE	Fecha de RCA de la entidad.
→	CAMPO_EPIGRAFEIAE	CADENA(30)	Epígrafe IAE.
→	CAMPO_EPIGRAFECNAE	CADENA(30)	Epígrafe CNAE.
→	CAMPO_COMENTARIOS	CADENA(100)	Comentarios.
→	CAMPO_OTROSDATOS	CADENA(1)	Indica si el interesado tiene otros datos → 'S' – Sí → 'N' – No
→	CAMPO_TIPOORGZ	CADENA(1)	Tipo de organización.
→	CAMPO_ANAGRAMAFISCAL	CADENA(20)	Anagrama fiscal del interesado
→	CAMPO_REFDATOCONT	NUMÉRICO	Identificador de los datos de contacto
→	CAMPO_MOTIVOBAJA	CADENA(100)	Motivo de la baja

Métodos para aplicación de los filtros y dónde se usa la clase

- insertarInteresado
- insertarInteresadoExpediente
- modificarInteresado
- obtenerInteresadoSimple
- obtenerInteresados

2.4.1.2.49 TrInteresadoDocumento

trewa.bd.trapi.trapiui.tpo.TrInteresadoDocumento

Clase que representa la información de los interesados en un documento del expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDOCEXP	TpoPK	Identificador del documento del expediente
→ REFTIPODOC	TpoPK	Identificador del tipo de documento
→ NOMBRETIPODOC	String	Nombre del tipo de documento
→ INTERESADOEXP	TrInteresadoExpediente	Interesado del expediente
→ RAZONINT	TrRazonInteres	Razón de interés.
→ OBSERVACIONES	String	Observaciones

→ **CONSENTIMIENTO** String

Indica si se da el consentimiento para el expediente y el documento en el que está interesado para realizar consultas a sus certificador.

Sus valores pueden ser:

- "S" - Sí
- "N" - No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDOCEXP	NUMÉRICO	Identificador del documento del expediente
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento
→ CAMPO_NOMBRETIPODOC	CADENA(50)	Nombre del tipo de documento
→ CAMPO_REFEXPEDIENTE	NUMÉRICO	Identificador del expediente.
→ CAMPO_REFINTERESADO	NUMÉRICO	Identificador del interesado
→ CAMPO_REFRAZONINTEXP	NUMÉRICO	Identificador de la razón de interés del interesado en el expediente.
→ CAMPO_REFRAZONINTDOC	NUMÉRICO	Identificador de la razón de interés.
→ CAMPO_OBSERVACIONES	CADENA(250)	Observaciones del interesado en el documento
→ CAMPO_TIPOIDENTIFICACION	CADENA(10)	Tipo de identificación
→ CAMPO_NUMIDENTIFICACION	CADENA(15)	Número de identificación.
→ CAMPO_APELLIDO1	CADENA(50)	Primer apellido del interesado.
→ CAMPO_APELLIDO2	CADENA(50)	Segundo apellido del interesado.
→ CAMPO_NOMBRE	CADENA(100)	Nombre del interesado.
→ CAMPO_SEXO	CADENA(1)	Sexo del interesado. → 'F' – Femenino → 'M' – Masculino
→ CAMPO_DIGCONTROL	CADENA(1)	Dígito de control para el número de identificación.
→ CAMPO_CIWAINTE	CADENA(15)	Código de identificación w@ndA del interesado
→ CAMPO_FECHANACIMIENTO	DATE	Fecha de nacimiento.
→ CAMPO_FECHABAJA	DATE	Fecha de baja en el sistema.
→ CAMPO_NUMREGMERCANTIL	CADENA(30)	Número del registro mercantil.
→ CAMPO_FECHARCA	DATE	Fecha de RCA de la entidad.
→ CAMPO_EPIGRAFEIAE	CADENA(30)	Epígrafe IAE.
→ CAMPO_EPIGRAFECNAE	CADENA(30)	Epígrafe CNAE.
→ CAMPO_COMENTARIOS	CADENA(100)	Comentarios.
→ CAMPO_TIPOORGANIZACION	CADENA(1)	Tipo de organización.
→ CAMPO_ANAGRAMAFISCAL	CADENA(20)	Anagrama fiscal del interesado
→ CAMPO_REFDATOCONTACTO	NUMÉRICO	Identificador de los datos de contacto
→ CAMPO_ABREVIATURAZONINTEXP	CADENA(10)	Abreviatura de la razón de interés del interesado en el expediente.
→ CAMPO_DESCRACIONINTEXP	CADENA(100)	Descripción de la razón de interés del interesado en el expediente.

EXP		
→ CAMPO_OBSOLETORAIN EXP	CADENA(1)	Indica si la razón de interés del interesado en el expediente está obsoleta → ' S ' – Sí → ' N ' – No
→ CAMPO_CODWANDARAZ ONINTEXP	CADENA(10)	Valor en w@ndA de la razón de interés del interesado en el expediente.
→ CAMPO_ABREVRACIONIN TDOC	CADENA(10)	Abreviatura de la razón de interés del interesado en el documento.
→ CAMPO_DESCRAZONINT DOC	CADENA(100)	Descripción de la razón de interés del interesado en el documento.
→ CAMPO_OBSOLETORAIN DOC	CADENA(1)	Indica si la razón de interés del interesado en el documento está obsoleta → ' S ' – Sí → ' N ' – No
→ CAMPO_CODWANDARAZ ONINTDOC	CADENA(10)	Valor en w@ndA de la razón de interés del interesado en el documento.
→ CAMPO_INFORMADO	CADENA(1)	Indica si se ha informado al bus del expediente Sus valores pueden ser: - " S " - Sí - " N " – No
→ CAMPO_BORRADO	CADENA(1)	Indica si el documento está borrado Sus valores pueden ser: - " S " - Sí - " N " – No
→ CAMPO_NUMEXP	CADENA(30)	Número del expediente
→ CAMPO_TITULOEXP	CADENA(250)	Título del expediente
→ CAMPO_OBSERVACIONES	CADENA(250)	Observaciones del expediente
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo al que pertenece el expediente
→ CAMPO_REFORGENVIA	NUMÉRICO	Identificador del organismo que envía el expediente
→ CAMPO_URLWANDA	CADENA(255)	Url del expediente en w@ndA
→ CAMPO_OTROSDATOS	CADENA(1)	Indica si el expediente tiene otros datos Sus valores pueden ser: - " S " - Sí - " N " – No
→ CAMPO_FECHAARCHIVO	DATE	Fecha de archivo del expediente
→ CAMPO_REFCOMPONENTE	NUMÉRICO	Identificador del componente donde está archivado el expediente
→ CAMPO_PROPIETARIOEXP	CADENA(10)	Usuario propietario del expediente
→ CAMPO_CONSENTIMIENTO	CADENA(1)	Indica si se da el consentimiento para el expediente y el documento en el que está interesado para realizar consultas a sus certificador. Sus valores pueden ser: - " S " - Sí - " N " – No

Métodos para aplicación de los filtros y dónde se usa la clase

- insertarInteresadoDocumento
- modificarInteresadoDocumento
- obtenerInteresadoDocumento

2.4.1.2.50 TrInteresadoExpediente

trewa.bd.trapi.trapiui.tpo.TrInteresadoExpediente

Clase que representa la información de los interesados en un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFEXPEDIENTE	TpoPK	Identificador del expediente
→ INTERESADO	TrInteresado	Interesado del expediente.
→ RAZONINT	TrRazonInteres	Razón de interés.
→ OBSERVACIONES	String	Observaciones del interesado en el expediente.
→ REFDATOCONT	TpoPK	Identificador de los datos de contacto
→ EXPEDIENTE	TrExpediente	Expediente.
→ NOTIFICA	String	Notificación telemática como medio preferente.
→ DATOCONTACTON OTIF	TpoPK	Notificación para el dato de contacto
→ FBAJA	Timestamp	Fecha de la baja del interesado en el expediente
→ MOTIVOBAJA	String	Motivo de la baja del interesado en el expediente

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFEXPEDIENTE	NUMÉRICO	Identificador del expediente.
→ CAMPO_REFINTERESADO	NUMÉRICO	Identificador del interesado.
→ CAMPO_REFRAZONINT	NUMÉRICO	Identificador de la razón de interés.
→ CAMPO_OBSERVACIONES	CADENA(250)	Observaciones del interesado en el expediente
→ CAMPO_REFDATOCONT	NUMÉRICO	Identificador de los datos de contacto
→ CAMPO_NOTIFICA	BOOLEAN	Notificación telemática como medio preferente, 'N' por defecto → 'S' - Sí → 'N' - No
→ CAMPO_CODTIPOIDENT	CADENA(10)	Tipo de identificación
→ CAMPO_NUMIDENT	CADENA(15)	Número de identificación.
→ CAMPO_APELLIDO1	CADENA(50)	Primer apellido del interesado.



→	CAMPO_APELLIDO2	CADENA(50)	Segundo apellido del interesado.
→	CAMPO_NOMBRE	CADENA(100)	Nombre del interesado.
→	CAMPO_SEXO	CADENA(1)	Sexo del interesado. → ' F ' – Femenino → ' M ' – Masculino
→	CAMPO_DIGCONTROL	CADENA(1)	Dígito de control para el número de identificación.
→	CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA del interesado
→	CAMPO_FECHANACIM	DATE	Fecha de nacimiento.
→	CAMPO_FECHABAJA	DATE	Fecha de baja en el sistema.
→	CAMPO_NUMREGMER	CADENA(30)	Número del registro mercantil.
→	CAMPO_FECHARCA	DATE	Fecha de RCA de la entidad.
→	CAMPO_EPIGRAFEIAE	CADENA(30)	Epígrafe IAE.
→	CAMPO_EPIGRAFECNA	CADENA(30)	Epígrafe CNAE.
	E		
→	CAMPO_COMENTARIOS	CADENA(100)	Comentarios.
→	CAMPO_REFTIPOORGZ	CADENA(1)	Tipo de organización.
→	CAMPO_ANAGRAMAFISCAL	CADENA(20)	Anagrama fiscal del interesado
→	CAMPO_ABREVIATURA	CADENA(10)	Abreviatura de la razón de interés del interesado en el expediente.
→	CAMPO_DESCRIPCION	CADENA(100)	Descripción de la razón de interés del interesado en el expediente.
→	CAMPO_OBSOLETO	CADENA(1)	Indica si la razón de interés del interesado en el expediente está obsoleta → ' S ' – Sí → ' N ' – No
→	CAMPO_CODWANDA	CADENA(10)	Valor en w@ndA de la razón de interés del interesado en el expediente.
→	CAMPO_INFORMADO	CADENA(1)	Indica si se ha informado al bus del expediente Sus valores pueden ser: - " S " - Sí - " N " – No
→	CAMPO_BORRADO	CADENA(1)	Indica si el expediente está borrado Sus valores pueden ser: - " S " - Sí - " N " – No
→	CAMPO_NUMEXP	CADENA(30)	Número del expediente
→	CAMPO_TITULOEXP	CADENA(250)	Título del expediente
→	CAMPO_OBSERVACIONESEXP	CADENA(250)	Observaciones del expediente
→	CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo al que pertenece el expediente
→	CAMPO_REFORGENVIA	NUMÉRICO	Identificador del organismo que envía el expediente
→	CAMPO_URLWANDA	CADENA(255)	Url del expediente en w@ndA
→	CAMPO_OTROSDATOS	CADENA(1)	Indica si el expediente tiene otros datos Sus valores pueden ser: - " S " - Sí - " N " – No
→	CAMPO_FECHAARCHIVO	DATE	Fecha de archivo del expediente

→	CAMPO_REFCOMPONENTE	NUMÉRICO	Identificador del componente donde está archivado el expediente
→	CAMPO_OTROSDATOSINTE	CADENA(1)	Indica si el interesado tiene otros datos Sus valores pueden ser: - "S" - Sí - "N" - No
→	CAMPO_PROPIETARIOEXP	CADENA(10)	Usuario propietario del expediente
→	CAMPO_F_BAJA	DATE	Fecha de baja del interesado asociado al expediente
→	CAMPO_IDENTEMPRESA	CADENA(15)	Identificador de empresa del interesado
→	CAMPO_IDENTENI	CADENA(55)	Identificador ENI
→	CAMPO_MOTIVO_BAJA	CADENA(100)	Motivo de la baja asociado al interesado del expediente
→	CAMPO_OBSERVACIONESEX	CADENA(250)	Observaciones del expediente
→	CAMPO_RAZONSOCIAL	CADENA(150)	Razón social del interesado
→	CAMPO_RPA_PROCEDIMIENTO	CADENA(20)	RPA del procedimiento al que pertenece el expediente
→	CAMPO_TIPOIDENTEMPRESA	CADENA(10)	Tipo del documento de empresa del interesado
→	CAMPO_CSV_EXP	CADENA(60)	CSV ENI
→	CAMPO_DATOSCONTACTONOTIF	NUMÉRICO	Identificador del dato de contacto de la notificación

Métodos para aplicación de los filtros y uso de la clase

- importarExpediente
- insertarInteresadoExpediente
- modificarInteresadoExpediente
- obtenerInteresadoExpediente
- explorarExpedientes
- obtenerInteresadoExpedienteCC
- obtenerInteresadosExpedienteDadoDeBaja

2.4.1.2.51 TrLimiteRelacion

trewa.bd.trapi.trapiui.tpo.TrLimiteRelacion

Clase que representa un límite de una relación entre interesados para particularizar la representación en una metafase, fase o expediente de un procedimiento.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	REFLIMITERELA	TpoPK	Identificador del límite de la relación
→	REFRELACIONINTE	TpoPK	Identificador de la relación entre interesados
→	FECHAINIVIG	Timestamp	Fecha de inicio de vigencia del límite

→	FECHAFINVIG	Timestamp	Fecha de fin de vigencia del límite
→	REFMETAFASE	TpoPK	Identificador de la metafase. Obligatorio si no se indica fase o expediente.
→	REFFASE	TpoPK	Identificador de la fase. Obligatorio si no se indica metafase o expediente..
→	REFEXPEDIENTE	TpoPK	Identificador del expediente. Obligatorio si no se indica metafase o fase.

Campos definidos para filtrados y ordenación

	<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_REFLIMITEREL A	NUMÉRICO	Identificador del límite de la relación
→	CAMPO_REFRELACIONINTE	NUMÉRICO	Identificador de la relación entre interesados
→	CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia del límite
→	CAMPO_FECHAFINVIG	DATE	Fecha de fin de vigencia del límite
→	CAMPO_REFMETAFASE	NUMÉRICO	Identificador de la metafase
→	CAMPO_REFFASE	NUMÉRICO	Identificador de la fase
→	CAMPO_REFEXPEDIENTE	NUMÉRICO	Identificador del expediente

Métodos para aplicación de los filtros

→ obtenerLimitesRelacion

2.4.1.2.52 TrMensaje

trewa.bd.trapi.trapiui.tpo.TrMensaje

Clase que representa la información referente a mensajes.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	REFMENSAJE	TpoPK	Identificador del mensaje
→	TEXTOMENSAJE	String	Texto del mensaje
→	LEIDO	String	Indica si el mensaje ha sido o no leído. → ' S ' – El mensaje ha sido leído → ' N ' – El mensaje no ha sido leído
→	FECHA	Timestamp	Fecha en la que se envió el mensaje
→	PRIORIDAD	String	Prioridad del mensaje → ' A ' – Prioridad alta → ' M ' – Prioridad media

→	USUARIOENV	String	→ 'B' – Prioridad baja
→	NOMBREUSUENV	String	Usuario origen del mensaje
→	USUARIOREC	String	Nombre y apellidos del usuario que envía el mensaje
→	NOMBREUSUREC	String	Usuario destino del mensaje
→	NOMBREUSUREC	String	Nombre y apellidos del usuario que recibe el mensaje
→	EXPEDIENTE	TrExpediente	Expediente al que está asociado

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFMENSAJE	NUMÉRICO	Identificador del mensaje
→ CAMPO_TEXTOMENSAJE	CADENA(250)	Texto del mensaje
→ CAMPO_LEIDO	CADENA	Indica si el mensaje ha sido o no leído. → 'S' – El mensaje ha sido leído → 'N' – El mensaje no ha sido leído
→ CAMPO_FECHA	DATE	Fecha en la que se envió el mensaje
→ CAMPO_PRIORIDAD	CADENA(1)	Prioridad del mensaje → 'A' – Prioridad alta → 'M' – Prioridad media → 'B' – Prioridad baja
→ CAMPO_USUARIOENV	CADENA(10)	Usuario origen del mensaje
→ CAMPO_USUARIOREC	CADENA(10)	Usuario destino del mensaje
→ CAMPO_REFEXP	NUMÉRICO	Identificador del expediente al que está asociado
→ CAMPO_NOMBREUSUENV	CADENA(98)	Nombre y apellidos del usuario que envía el mensaje
→ CAMPO_NOMBREUSUREC	CADENA(98)	Nombre y apellidos del usuario que recibe el mensaje
→ CAMPO_INFORMADO	CADENA(1)	Indica si se ha informado al bus del expediente Sus valores pueden ser: - "S" - Sí - "N" – No
→ CAMPO_NUMEXP	CADENA(30)	Número del expediente
→ CAMPO_TITULOEXP	CADENA(250)	Título del expediente
→ CAMPO_OBSERVACIONES	CADENA(250)	Observaciones del expediente
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo al que pertenece el expediente
→ CAMPO_REFORGENVIA	NUMÉRICO	Identificador del organismo que envía el expediente
→ CAMPO_URLWANDA	CADENA(255)	Url del expediente en w@ndA
→ CAMPO_OTROS DATOS	CADENA(1)	Indica si el expediente tiene otros datos Sus valores pueden ser: - "S" - Sí - "N" – No
→ CAMPO_FECHAARCHIVO	DATE	Fecha de archivo del expediente
→ CAMPO_REFCOMPONENTE	NUMÉRICO	Identificador del componente donde está archivado el expediente

→ **CAMPO_PROPIETARIOE** CADENA(10) Usuario propietario del expediente
XP

Métodos para aplicación de los filtros

- obtenerMensajesUsuario
- obtenerMensajesEnviados

2.4.1.2.53 TrMensajeCondicionAccion

trewa.bd.trapi.trapiui.tpo.TrMensajeCondicionAccion

Clase que representa la información referente a un mensaje asociado a una acción, condición o un mensaje de aviso.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	NOMBRE	String	Nombre de la condición o acción
→	DESCRIPCION	String	Descripción de la condición o acción
→	MENSAJE	String	Resultado de evaluar la condición o realizar la acción
→	TIPO	String	Indica si el mensaje corresponde a una condición, a una acción o a un mensaje de aviso. → 'C' – El mensaje corresponde a una condición → 'A' – El mensaje corresponde a una acción → 'W' – Se trata de un mensaje de aviso → evaluarCondAccDocuImpl → evaluarCondAccOtrasTareasImpl → evaluarCondAccTransImpl → evaluarCondicionComplejaDocu → evaluarCondicionComplejaOtrasTareas → evaluarCondicionComplejaTrans

Métodos para aplicación de los filtros

2.4.1.2.54 TrMetadato

trewa.bd.trapi.trapiui.tpo.TrMetadato

Clase que representa la información referente a un metadato.

El modelo de referencia w@ndA define un metadato como “el conjunto de tipos de datos estandarizados que pueden ser utilizados en documentos y expedientes electrónicos”.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoria
→ REFMETADATO	TpoPK	Identificador del metadato
→ NOMBRE	String	Nombre del metadato
→ DESCRIPCION	String	Descripción del metadato
→ EQUIVALENCIAEE MGDE	String	Equivalencia con los Metadatos para la Gestión del Documento Electrónico.
→ ESQUEMAVALOR	String	Esquema de los valores aceptados
→ PAQUETE	String	Nombre del paquete que contiene la función almacenada o método JAVA
→ NOMBREFUNCION	String	Nombre de la función almacenada que representa al metadato o nombre del método en el caso de llamada JAVA
→ PAQUETE OBLIGAT ORIEDAD	String	Nombre del paquete que contiene la función almacenada o método JAVA de la condición de obligatoriedad del metadato
→ NOMBREFUNCION OBLIGATORIEDAD	String	Nombre de la función almacenada que representa al metadato o nombre del método en el caso de llamada a JAVA, de la condición de obligatoriedad
→ DOCUMENTO	String	Indica si se trata de un metadato para un documento → “S” – Sí → “N” – No
→ EXPEDIENTE	String	Indica si se trata de un metadato para un expediente → “S” – Sí → “N” – No
→ PARAMREFEXP	String	Indica si se le pasa el id del expediente al metadato. → “S” – Sí → “N” – No
→ PARAMREFDOC	String	Indica si se le pasa el id del documento al metadato. → “S” – Sí → “N” – No
→ PARAMREFTRAN	String	Indica si se le pasa el id de la transición al metadato. → “S” – Sí → “N” – No
→ PARAMREFDOCPE R	String	Indica si se le pasa el id del documento permitido al metadato. → “S” – Sí → “N” – No
→ PARAMREFEXPXFA S	String	Indica si se le pasa el id del expediente en fase al metadato. → “S” – Sí → “N” – No
→ PARAMREFDEFPRO C	String	Indica si se le pasa el id del procedimiento al metadato. → “S” – Sí → “N” – No
→ PARAMFECHA	String	Indica si se le pasa la fecha al metadato. → “S” – Sí

- | | | | |
|---|-----------------------------|--------|--|
| → | PARAMUSUARIO | String | → "N" – No
Indica si se le pasa el usuario al metadato.
→ "S" – Sí
→ "N" – No |
| → | PARAMREFFASE | String | Indica si se le pasa el id de la fase al metadato.
→ "S" – Sí
→ "N" – No |
| → | PARAMREFTIPODO
C | String | Indica si se le pasa el id del tipo documento al metadato.
→ "S" – Sí
→ "N" – No |

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICACION	DATE	Fecha de modificación del registro
→ CAMPO_REFMETADATO	NUMÉRICO	Identificador de la metadato
→ CAMPO_NOMBRE	CADENA(25)	Nombre del metadato
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del metadato
→ CAMPO_EQUIVALENCIAEEMGDE	CADENA(250)	Equivalencia con los Metadatos para la Gestión del Documento Electrónico.
→ CAMPO_ESQUEMAVALOR	CADENA(250)	Esquema de los valores aceptados
→ CAMPO_PAQUETE	CADENA(100)	Nombre del paquete que contiene la función almacenada o método JAVA
→ CAMPO_NOMBREFUNCION	CADENA(50)	Nombre de la función almacenada que representa al metadato o nombre del método en el caso de llamada JAVA
→ CAMPO_PAQUETEOBLIGATORIEDAD	CADENA(100)	Nombre del paquete que contiene la función almacenada o método JAVA de la condición de obligatoriedad del metadato
→ CAMPO_NOMBREFUNCIONOBLIGATORIEDAD	CADENA(50)	Nombre de la función almacenada que representa al metadato o nombre del método en el caso de llamada a JAVA, de la condición de obligatoriedad
→ CAMPO_DOCUMENTO	CADENA(1)	Indica si se trata de un metadato para un documento → "S" – Sí → "N" – No
→ CAMPO_EXPEDIENTE	CADENA(1)	Indica si se trata de un metadato para un expediente → "S" – Sí → "N" – No
→ CAMPO_PARAMREFEXIP	CADENA(1)	Indica si se le pasa el id del expediente al metadato. → "S" – Sí → "N" – No
→ CAMPO_PARAMREFDOC	CADENA(1)	Indica si se le pasa el id del documento al metadato. → "S" – Sí → "N" – No
→ CAMPO_PARAMREFTR	CADENA(1)	Indica si se le pasa el id de la transición al metadato.



AN		→ "S" – Sí → "N" – No
→ CAMPO_PARAMREFDO	CADENA(1)	Indica si se le pasa el id del documento permitido al metadato.
CPER		→ "S" – Sí → "N" – No
→ CAMPO_PARAMREFEX	CADENA(1)	Indica si se le pasa el id del expediente en fase al metadato.
PXFAS		→ "S" – Sí → "N" – No
→ CAMPO_PARAMREFDE	CADENA(1)	Indica si se le pasa el id del procedimiento al metadato.
FPROC		→ "S" – Sí → "N" – No
→ CAMPO_PARAMFECHA	CADENA(1)	Indica si se le pasa la fecha al metadato.
		→ "S" – Sí → "N" – No
→ CAMPO_PARAMUSUAR	CADENA(1)	Indica si se le pasa el usuario al metadato.
IO		→ "S" – Sí → "N" – No
→ CAMPO_PARAMREFFA	CADENA(1)	Indica si se le pasa el id de la fase al metadato.
SE		→ "S" – Sí → "N" – No
→ CAMPO_PARAMREFTIP	CADENA(1)	Indica si se le pasa el id del tipo documento al metadato.
ODOC		→ "S" – Sí → "N" – No
→ INIT_ROW	NUMERICO	Indica el primer registro a devolver
→ MAX_ROW	NUMERICO	Indica el número máximo de filas a devolver

Métodos para aplicación de los filtros → obtenerMetadato

2.4.1.2.55 TrMetafase

trewa.bd.trapi.trapiui.tpo.TrMetafase

Clase que representa la información referente a una metafase.

El modelo de referencia w@ndA define una metafase como "el conjunto de fases dentro de un procedimiento que comparten una serie de características comunes que permiten al Tramitador dar un tratamiento diferenciado del resto pero común a todas ellas".

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoria
→ REFMETAFASE	TpoPK	Identificador de la metafase
→ NOMBRE	String	Nombre de la metafase
→ DESCRIPCION	String	Descripción de la metafase
→ ORDEN	long	Orden
→ STMA	TrSistema	Sistema
→ INFORMARBUS	String	Indica si se informa al bus al entrar en la metafase → 'S' – Sí → 'N' – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFMETAFAS E	NUMÉRICO	Identificador de la metafase.
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la metafase.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la metafase.
→ CAMPO_ORDEN	NUMÉRICO	Orden no riguroso de la metafase dentro del procedimiento en el que aparecen.
→ CAMPO_INFORMARBUS	CADENA(1)	Indica si se informa al bus de la entrada en la metafase. → "S" – Sí → "N" – No
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICA	DATE	Fecha de modificación del registro

- explorarExpedientesSimple
- explorarExpedientes
- obtenerDatosFase
- obtenerFases
- obtenerFasesExpediente
- obtenerFasesFinTransicion
- obtenerFasesFinTransicion
- obtenerFasesProcedimiento
- obtenerMetafases
- obtenerMetafasesProcedimiento
- obtenerTransicionesDefProcedimiento

Métodos para aplicación de los filtros y dónde se usa

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.1.2.56 TrModificacionCaducidadExpediente

trewa.bd.trapi.trapiui.tpo.TrModificacionCaducidadExpediente

Clase que representa la información referente a una modificación de la caducidad de un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ TIPO	String	Tipo de la modificación → 'A' – Ampliación → 'R' – Reducción → 'S' – Suspensión
→ UNIDAD	String	Unidad de medida de tiempo → 'D' – Días → 'M' – Meses → 'A' – Años
→ NUMUNIDADES	long	Número de unidades que se cuentan (nº de días, nº de años,...)
→ FECHA	Timestamp	Fecha en la que se produjo la modificación sobre la caducidad
→ FECHAFINAL	Timestamp	Fecha de finalización para las modificaciones del tipo "suspensión". Es la fecha en la que sea reanuda la cuenta del tiempo.
→ USUARIO	String	Usuario que hizo la modificación

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_TIPO	CADENA(1)	Tipo de la modificación → 'A' – Ampliación → 'R' – Reducción → 'S' – Suspensión
→ CAMPO_UNIDAD	CADENA(1)	Unidad de medida de tiempo → 'D' – Días → 'M' – Meses → 'A' – Años
→ CAMPO_NUMUNIDADES	NUMÉRICO	Número de unidades que se cuentan (nº de días, nº de años,...)
→ CAMPO_FECHA	DATE	Fecha en la que se produjo la modificación sobre la caducidad
→ CAMPO_FECHAFINAL	DATE	Fecha de finalización para las modificaciones del tipo "suspensión". Es la fecha en la que sea reanuda la cuenta del tiempo.
→ CAMPO_USUARIO	CADENA(10)	Usuario que hizo la modificación

Métodos para aplicación de los filtros

→ obtenerModificacionesCaducidadExpediente

2.4.1.2.57 TrMunicipio

trewa.bd.trapi.trapiui.tpo.TrMunicipio

Clase que representa la información de un municipio

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODMUNICIPIO	String	Código del municipio.
→ PROVINCIA	TrProvincia	Provincia a la que pertenece el municipio.
→ NOMBRE	String	Nombre del municipio.
→ DESCRIPCION	String	Descripción del municipio.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODMUNICIPIO	CADENA(3)	Código del municipio.
→ CAMPO_CODPROVINCIA	CADENA(2)	Provincia a la que pertenece el municipio.
→ CAMPO_NOMBRE	CADENA(32)	Nombre del municipio.
→ CAMPO_DESCRIPCION	CADENA(64)	Descripción del municipio.

Métodos para aplicación de los filtros

- obtenerMunicipios
- obtenerDatosContactoInteresado
- obtenerDatosContactoo
- obtenerEmpleado
- obtenerMunicipios
- obtenerOrganismo
- obtenerOrganismosHijos
- obtenerOrganismoTramitaExpediente

2.4.1.2.58 TrNormativaProcedimiento

trewa.bd.trapi.trapiui.tpo.TrNormativaProcedimiento

Clase que representa las distintas normativas por las que se rige un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento.
→ NORMATIVA	TrNormativa	Normativa.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_REFNORMATIVA	NUMÉRICO	Identificador de la normativa.
→ CAMPO_REFTIPONORMA	NUMÉRICO	Identificador del tipo de normativa.
→ CAMPO_REFTIPOPUBLI	NUMÉRICO	Identificador del tipo de publicación.
→ CAMPO_DESCNORMATIVA	CADENA(250)	Descripción de la normativa.
→ CAMPO_FECHAVIGOR	DATE	Fecha de entrada en vigor de la normativa.
→ CAMPO_TITULO	CADENA(50)	Título de la normativa
→ CAMPO_NUMERO	NUMÉRICO	Número de la normativa.
→ CAMPO_FECHAPUBLI	DATE	Fecha de publicación de la normativa.
→ CAMPO_NUMPUBLI	NUMÉRICO	Número de publicación de la normativa.
→ CAMPO_CODWANDA	CADENA(15)	Valor en w@ndA de la normativa
→ CAMPO_ANYO	NUMÉRICO	Año de la normativa.
→ CAMPO_REFAMBITOLEY	NUMÉRICO	Identificador del ámbito de la ley de la normativa.

Métodos para aplicación de los filtros

→ obtenerNormativasProcedimiento

2.4.1.2.59 TrNotificacionEnvio

trewa.bd.trapi.trapiui.tpo.TrNotificacionEnvio

Clase que representa la información de los envíos de las notificaciones.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFNOTENV	TpoPK	Identificador del envío
→ REFNOTREM	TpoPK	Identificador de la remesa
→ FECHA	Timestamp	Fecha del envío
→ MEDIO	String	Medio de realización del envío
→ RECHAZO	String	Motivo de rechazo del envío
→ HASHNOTIF	String	Código HASH del envío
→ FECHAESTADO	Timestamp	Fecha del cambio de estado
→ ESTADO	Integer	Estado del envío
→ ASUNTO	String	Asunto de la notificación
→ CUERPO	String	Cuerpo de la notificación
→ OBSERVACIONES	String	Observaciones del envío de notificación

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFNOTENV	NUMÉRICO	Identificador del envío
→ CAMPO_REFNOTREM	NUMÉRICO	Identificador de la remesa
→ CAMPO_FECHAENVIO	DATE	Fecha del envío
→ CAMPO_MEDIO	CADENA(100)	Medio de realización del envío
→ CAMPO_RECHAZO	CADENA(100)	Motivo de rechazo del envío
→ CAMPO_HASHNOTIF	CADENA(100)	Código HASH del envío
→ CAMPO_FECHAESTADO	DATE	Fecha del cambio de estado
→ CAMPO_ESTADO	CADENA(100)	Estado del envío
→ CAMPO_ASUNTO	CADENA(100)	Asunto de la notificación
→ CAMPO_CUERPO	CADENA(250)	Cuerpo de la notificación
→ CAMPO_OBSERVACIONES	CADENA(100)	Observaciones de la notificación
→ CAMPO_REFEXP	NUMÉRICO	Identificador del expediente
→ CAMPO_REFINTE	NUMÉRICO	Identificador del interesado
→ CAMPO_REFDOCEXP	NUMÉRICO	Identificador del documento expediente
→ CAMPO_REFRAZINTDOC	NUMÉRICO	Identificador de la razón interés documento
→ CAMPO_REFRAZINTEXP	NUMÉRICO	Identificador de la razón interés expediente

Métodos para aplicación de los filtros y dónde se usa

- checkRemesa
- clonarDocumentExpediente
- modificarNotificacionInteresado
- notificaDocumentosInteresados
- obtenerEnvioNotificacion

2.4.1.2.60 TrOrganismo

trewa.bd.trapi.trapiui.tpo.TrOrganismo

Clase que representa la información de los distintos organismos o unidades orgánicas.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	AUDITORIA	TrAuditoria	Datos de auditoria
→	REFORGANISMO	TpoPK	Identificador del organismo
→	CODORG	String	Código del organismo
→	NOMBRE	String	Nombre del organismo
→	DESCRIPCION	String	Descripción del organismo
→	TIPO	String	Tipo de organismo → ' DP ' – Delegación Provincial → ' SC ' – Servicios Centrales → ' CE ' – Centro Adscrito → ' EE ' – Empresa Externa
→	DATOSCONTACTO	TrDatosContacto(*)	Datos de contacto
→	CIWA	String	Código de identificación w@ndA.
→	IDARIES	String	Identificador de aries
→	IDENTIFICADOR	String	Número de identificación
→	DIGCONTROL	String	Dígito de control
→	FECHAINIVIG	Timestamp	Fecha de inicio de vigencia del organismo
→	FECHAFINIVIG	Timestamp	Fecha de fin de vigencia del organismo
→	REFORGPADRE	TpoPK	Identificador del organismo padre
→	TIPOORG	TrTipoOrganismo	Tipo de Organismo
→	TIPOORGZ	TrTipoOrganizacion	Tipo de Organización
→	ORDENPRELACION	String	Orden de relación del organismo

→ **DIR3** String Código de organismo en el Directorio Común de Unidades Orgánicas y Oficinas

(*) Se reutiliza la clase trewa.bd.trapi.trapiui.tpo.TrDatosContacto para devolver los datos de contacto del organismo aunque en el modelo de datos no existe una relación entre organismos y datos de contacto (propios del interesado). Por lo tanto algunos atributos no traerán valor, por ejemplo REFDATOCONT.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo
→ CAMPO_CODMUNICIPIO	CADENA(3)	Código del municipio
→ CAMPO_CODPROVINCIA	CADENA(2)	Código de la provincia
→ CAMPO_TIPOVIA	CADENA(10)	Tipo de vía
→ CAMPO_CODORG	CADENA(16)	Código del organismo
→ CAMPO_NOMBRE	CADENA(32)	Nombre del organismo
→ CAMPO_DESCRIPCION	CADENA(200)	Descripción del organismo
→ CAMPO_TIPO	CADENA(1)	Tipo de organismo → ' DP ' – Delegación Provincial → ' SC ' – Servicios Centrales → ' CE ' – Centro Adscrito → ' EE ' – Empresa Externa
→ CAMPO_NOMBREVIA	CADENA(64)	Nombre de la vía
→ CAMPO_NUMERO	NUMÉRICO	Número del domicilio
→ CAMPO_LETRA	CADENA(2)	Letra del domicilio
→ CAMPO_ESCALERA	CADENA(2)	Escalera del domicilio
→ CAMPO_PISO	NUMERICO	Piso del domicilio
→ CAMPO_PUERTA	CADENA(2)	Puerta del domicilio
→ CAMPO_CODPOSTAL	NUMERICO	Código postal del domicilio
→ CAMPO_TELEFONO	CADENA(25)	Número(s) de teléfono de contacto
→ CAMPO_TLFMOVIL	CADENA(25)	Número(s) de teléfono móvil
→ CAMPO_FAX	CADENA(25)	Número(s) de Fax
→ CAMPO_EMAIL	CADENA(64)	Correo electrónico
→ CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA
→ CAMPO_IDARIES	CADENA(5)	Identificador de aries
→ CAMPO_IDENTIFICADOR	CADENA(15)	Número de identificador
→ CAMPO_DIGCONTROL	CADENA(1)	Dígito de control
→ CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia del organismo
→ CAMPO_FECHAFINIVIG	DATE	Fecha de fin de vigencia del organismo
→ CAMPO_REFORGPADRE	NUMÉRICO	Identificador del organismo superior al que pertenece.
→ CAMPO_REFTIPOORG	NUMÉRICO	Identificador del tipo de organismo.
→ CAMPO_REFTIPOORGZ	NUMÉRICO	Identificador del tipo de organización.
→ CAMPO_ORDENPRELACION	CADENA(15)	Orden de relación del organismo
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICA	DATE	Fecha de modificación del registro

→ **CAMPO_DIR3** CADENA(10) Código de organismo en el Directorio Común de Unidades Orgánicas y Oficinas

- crearExpediente
- explorarExpedientesSimple
- explorarExpedientes
- obtenerDefProcedimientosDefinidos
- obtenerEmpleado
- obtenerExpedientes
- obtenerExpedientesCaducados
- obtenerExpsRelacionInteresado
- obtenerInteresadoDocumento
- obtenerInteresadoExpediente
- obtenerInteresadoExpedienteCC
- obtenerInteresadoExpedienteDadoDeBaja
- obtenerMensajesUsuario
- obtenerOrganismos
- obtenerOrganismosExpediente
- obtenerOrganismosHijos
- obtenerOrganismosProcedimiento
- obtenerOrganismosTramitaExpediente
- obtenerProcedimientosUsuario
- obtenerPtoTrabOrganismo
- obtenerUsuarioAsignado

Métodos para aplicación de los filtros y dónde se usa

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.1.2.61 TrOrganismoExpediente

trewa.bd.trapi.trapiui.tpo.TrOrganismoExpediente

Clase que representa la relación entre un organismo y un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFORGEEXP	TpoPK	Identificador de la relación entre el organismo y el expediente
→ ORGANISMO	TrOrganismo	Organismo
→ REFEXPEDIENTE	TpoPK	Identificador del expediente
→ TIPOASOCIA	String	Tipo de asociación entre el organismo y el expediente
→ OBSERVACIONES	String	Observaciones
→ FECHAINIVIG	Timestamp	Fecha de inicio de vigencia
→ FECHAFINIVIG	Timestamp	Fecha de fin de vigencia

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFORGEEXP	NUMÉRICO	Identificador de la relación entre el organismo y el expediente
→ CAMPO_REFEXPEDIENTE	NUMÉRICO	Identificador del expediente
→ CAMPO_TIPOASOCIA	CADENA	Tipo de asociación entre el organismo y el expediente
→ CAMPO_OBSERVACIONES	CADENA	Observaciones
→ CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia
→ CAMPO_FECHAFINIVIG	DATE	Fecha de fin de vigencia
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo
→ CAMPO_CODMUNICIPIO	CADENA(3)	Código del municipio
→ CAMPO_CODPROVINCIA	CADENA(2)	Código de la provincia
→ CAMPO_TIPOVIA	CADENA(10)	Tipo de vía
→ CAMPO_CODORG	CADENA(16)	Código del organismo
→ CAMPO_NOMBREORG	CADENA(32)	Nombre del organismo
→ CAMPO_DESCRIPCIONORG	CADENA(200)	Descripción del organismo
→ CAMPO_TIPO	CADENA(1)	Tipo de organismo → 'DP' – Delegación Provincial → 'SC' – Servicios Centrales → 'CE' – Centro Adscrito → 'EE' – Empresa Externa
→ CAMPO_NOMBREVIA	CADENA(64)	Nombre de la vía
→ CAMPO_NUMERO	NUMÉRICO	Número del domicilio

→	CAMPO_LETRA	CADENA(2)	Letra del domicilio
→	CAMPO_ESCALERA	CADENA(2)	Escalera del domicilio
→	CAMPO_PISO	NUMERICO	Piso del domicilio
→	CAMPO_PUERTA	CADENA(2)	Puerta del domicilio
→	CAMPO_CODPOSTAL	NUMERICO	Código postal del domicilio
→	CAMPO_TELEFONO	CADENA(25)	Número(s) de teléfono de contacto
→	CAMPO_TLFMOVIL	CADENA(25)	Número(s) de teléfono móvil
→	CAMPO_FAX	CADENA(25)	Número(s) de Fax
→	CAMPO_EMAIL	CADENA(64)	Correo electrónico
→	CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA
→	CAMPO_IDARIES	CADENA(5)	Identificador de aries
→	CAMPO_IDENTIFICADOR	CADENA(15)	Número de identificador
→	CAMPO_DIGCONTROL	CADENA(1)	Dígito de control
→	CAMPO_FECHAINIVIGORG	DATE	Fecha de inicio de vigencia del organismo
→	CAMPO_FECHAFINVIGORG	DATE	Fecha de fin de vigencia del organismo
→	CAMPO_REFORGPADRE	NUMÉRICO	Identificador del organismo superior al que pertenece.
→	CAMPO_REFTIPOORG	NUMÉRICO	Identificador del tipo de organismo.
→	CAMPO_REFTIPOORGZ	NUMÉRICO	Identificador del tipo de organización.
→	CAMPO_ORDENPRELACION	CADENA(15)	Orden de relación del organismo

Métodos para aplicación de los filtros

→ obtenerOrganismosExpediente

2.4.1.2.62 TrOrganismoProcedimiento

trewa.bd.trapi.trapiui.tpo.TrOrganismoProcedimiento

Clase que representa la relación entre un organismo y un procedimiento.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	REFORGPROC	TpoPK	Identificador de la relación entre el organismo y el procedimiento
→	ORGANISMO	TrOrganismo	Organismo
→	REFDEFPROC	TpoPK	Identificador del organismo
→	TIPOASOCIA	String	Tipo de asociación entre el organismo y el procedimiento
→	OBSERVACIONES	String	Observaciones
→	FECHAINIVIG	Timestamp	Fecha de inicio de vigencia
→	FECHAFINVIG	Timestamp	Fecha de fin de vigencia

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFORGPROC	NUMÉRICO	Identificador de la relación entre el organismo y el procedimiento
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador del procedimiento
→ CAMPO_TIPOASOCIA	CADENA	Tipo de asociación entre el organismo y el procedimiento
→ CAMPO_OBSERVACIONES	CADENA	Observaciones
→ CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia
→ CAMPO_FECHAFINVIG	DATE	Fecha de fin de vigencia
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo
→ CAMPO_CODMUNICIPIO	CADENA(3)	Código del municipio
→ CAMPO_CODPROVINCIA	CADENA(2)	Código de la provincia
→ CAMPO_TIPOVIA	CADENA(10)	Tipo de vía
→ CAMPO_CODORG	CADENA(16)	Código del organismo
→ CAMPO_NOMBREORG	CADENA(32)	Nombre del organismo
→ CAMPO_DESCRIPCIONORG	CADENA(200)	Descripción del organismo
G		
→ CAMPO_TIPO	CADENA(1)	Tipo de organismo → 'DP' – Delegación Provincial → 'SC' – Servicios Centrales → 'CE' – Centro Adscrito → 'EE' – Empresa Externa
→ CAMPO_NOMBREVIA	CADENA(64)	Nombre de la vía
→ CAMPO_NUMERO	NUMÉRICO	Número del domicilio
→ CAMPO_LETRA	CADENA(2)	Letra del domicilio
→ CAMPO_ESCALERA	CADENA(2)	Escalera del domicilio
→ CAMPO_PISO	NUMERICO	Piso del domicilio
→ CAMPO_PUERTA	CADENA(2)	Puerta del domicilio
→ CAMPO_CODPOSTAL	NUMERICO	Código postal del domicilio
→ CAMPO_TELEFONO	CADENA(25)	Número(s) de teléfono de contacto
→ CAMPO_TLFMOVIL	CADENA(25)	Número(s) de teléfono móvil
→ CAMPO_FAX	CADENA(25)	Número(s) de Fax
→ CAMPO_EMAIL	CADENA(64)	Correo electrónico
→ CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA
→ CAMPO_IDARIES	CADENA(5)	Identificador de aries
→ CAMPO_IDENTIFICADOR	CADENA(15)	Número de identificador
→ CAMPO_DIGCONTROL	CADENA(1)	Dígito de control
→ CAMPO_FECHAINIVIGORG	DATE	Fecha de inicio de vigencia del organismo
→ CAMPO_FECHAFINVIGORG	DATE	Fecha de fin de vigencia del organismo
G		
→ CAMPO_REFORGPADRE	NUMÉRICO	Identificador del organismo superior al que pertenece.
→ CAMPO_REFTIPOORG	NUMÉRICO	Identificador del tipo de organismo.
→ CAMPO_REFTIPOORGZ	NUMÉRICO	Identificador del tipo de organización.
→ CAMPO_ORDENPRELACION	CADENA(15)	Orden de relación del organismo
N		

Métodos para aplicación de los filtros

→ obtenerOrganismosProcedimiento

2.4.1.2.63TrPagoTasa

trewa.bd.trapi.trapiui.tpo.TrPagoTasa

Clase que representa la información referente a la instancia del pago.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPAGOTASA	TpoPK	Identificador del pago.
→ ERROR_DOC	String	Mensaje de error del documento de pago.
→ ERROR_PAGO	String	Mensaje de error del pago.
→ ESTADO_PAGO	String	Estado del pago.
→ ESTADO_DOC	String	Estado del documento de pago.
→ FECHA_PAGO	Timestamp	Fecha del pago.
→ TIOPAGO_PROC	TrTipoPagoProcedimiento	Tipo de pago procedimiento.
→ IMPORTE	Float	Importe del pago.
→ REFINTERESADO	TpoPK	Identificador de l interesado.
→ NUMERO_DOC	String	Número del documento de pago.
→ CONCEPTO	String	Concepto del pago.
→ REFEXPEDIENTE	TpoPK	Identificador del expediente.
→ DOCUMENTO	byte[]	Documento de pago.
→ REFDOCEXPEDIENTE	TpoPK	Identificador del documento expediente.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPAGOTASA	NUMÉRICO	Identificador del pago.
→ CAMPO_ERRORDOC	CADENA(100)	Mensaje de error del documento de pago.
→ CAMPO_ERRORPAGO	CADENA(100)	Mensaje de error del pago.
→ CAMPO_ESTADOPAGO	CADENA(1)	Estado del pago.
→ CAMPO_ESTADODOC	CADENA(5)	Estado del documento de pago.
→ CAMPO_FECHAPAGO	NUMÉRICO	Fecha del pago.
→ CAMPO_REFTIPOPAGO	NUMÉRICO	Tipo de pago.
→ CAMPO_IMPORTE	NUMÉRICO	Importe del pago.
→ CAMPO_REFINTERESADO	NUMÉRICO	Identificador de l interesado.
→ CAMPO_NUMERODOC	CADENA(64)	Número del documento de pago.
→ CAMPO_CONCEPTO	CADENA(128)	Concepto del pago.
→ CAMPO_REFEXPEDIENTE	NUMÉRICO	Identificador del expediente.
→ CAMPO_REFDOCEXPEDIENTE	NUMÉRICO	Identificador del documento expediente.

Métodos para aplicación de los filtros

→ obtenerPagosTasas

2.4.1.2.64TrPais

trewa.bd.trapi.trapiui.tpo.TrPais

Clase que representa la información de un país

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODPAIS	String	Identificador del país
→ NOMBRE	String	Nombre del país
→ DESCRIPCION	String	Descripción del país

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODPAIS	CADENA(3)	Identificador del país
→ CAMPO_NOMBRE	CADENA(32)	Nombre del país
→ CAMPO_DESCRIPCION	CADENA(64)	Descripción del país

Métodos para aplicación de los filtros

→ obtenerDatosContactoInteresado

→ obtenerDatosContactoo

→ obtenerPais

2.4.1.2.65 TrParametro

trewa.bd.trapi.trapiui.tpo.TrParametro

Clase que representa la información referente a un parámetro.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPARAM	TpoPK	Identificador del parámetro
→ NOMBRE	String	Nombre del parámetro
→ DESCRIPCION	String	Descripción del parámetro
→ TIPO	String	Tipo de parámetro → ' C ' – Carácter → ' N ' – Numérico → ' F ' – Fecha
→ TAMANIO	long	Tamaño del parámetro
→ STMA	TrSistema	Sistema asociado al parámetro

Métodos para aplicación de los filtros

→ obtenerParametrosBloques

2.4.1.2.66 TrParametroBloque

trewa.bd.trapi.trapiui.tpo.TrParametroBloque

Clase que representa la información referente a un parámetro definido para un bloque.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ PARAMETRO	TrParametro	Parámetro
→ ORDEN	long	Orden del parámetro en la llamada al módulo

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_NOMBRE	CADENA(20)	Nombre del parámetro
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción del parámetro
→ CAMPO_TIPO	CADENA(1)	Tipo de parámetro → ' C ' – Carácter → ' N ' – Numérico → ' F ' – Fecha
→ CAMPO_TAMANIO	NUMÉRICO	Tamaño del parámetro

- **CAMPO_ORDEN** NUMÉRICO Orden del parámetro en la llamada al módulo
- **CAMPO_REFSTMA** NUMÉRICO Identificador del sistema asociado al parámetro

Métodos para aplicación de los filtros

→ obtenerParametrosBloque

2.4.1.2.67 TrParrafo

trewa.bd.trapi.trapiui.tpo.TrParrafo

Clase que representa la información referente a un párrafo de un documento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPARRDOC	TpoPK	Identificador del párrafo del documento del expediente
→ ETIQUETA	String	Etiqueta del párrafo del documento
→ PARRAFO	String	Texto del párrafo
→ ORDEN	long	Orden del párrafo
→ ALINEACION	String	Alineación del párrafo → “ I ” – A la izquierda → “ C ” – Centrada → “ D ” – A la derecha → “ J ” – Justificada
→ ESTILO	String	Estilo del párrafo
→ ESTILOETIQ	String	Estilo de la etiqueta del párrafo
→ EDITABLE	String	Indica si el párrafo es editable o no → ‘ S ’ – El párrafo es editable → ‘ N ’ – El párrafo no es editable
→ TIOPARRAFO	TrTipoParrafo	Tipo de párrafo
→ IMAGEN	byte[]	Imagen del párrafo
→ FORMATO	String	Tipo de formato (tipo mime)
→ NOMBREFICHERO	String	Nombre del fichero

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPARRDOC	NUMÉRICO	Identificador del párrafo del documento del expediente
→ CAMPO_ETIQUETA	CADENA(15)	Etiqueta del párrafo del documento
→ CAMPO_PARRAFO	CADENA(4000)	Texto del párrafo
→ CAMPO_ORDEN	NUMÉRICO	Orden del párrafo
→ CAMPO_ALINEACION	CADENA(1)	Alineación del párrafo → “ I ” – A la izquierda

→ CAMPO_ESTILO	CADENA(8)	→ “ C ” – Centrada
→ CAMPO_ESTILOETIQ	CADENA(8)	→ “ D ” – A la derecha
→ CAMPO_EDITABLE	CADENA(1)	→ “ J ” – Justificada
		Estilo del párrafo
		Estilo de la etiqueta del párrafo
		Indica si el párrafo es editable o no
		→ ‘ S ’ – El párrafo es editable
		→ ‘ N ’ – El párrafo no es editable
→ CAMPO_REFTIPOPARR	NUMÉRICO	Identificador del tipo de párrafo
→ CAMPO_ABREVTIPOPARR	CADENA(15)	Abreviatura del tipo de párrafo
R		
→ CAMPO_DESCTIPOPARR	CADENA(250)	Descripción del tipo de párrafo
→ CAMPO_FORMATO	CADENA(128)	Tipo de formato (tipo mime)
→ CAMPO_NOMBREFICHERO	CADENA(64)	Nombre del fichero

Métodos para aplicación de los filtros

- convertirParrafoTipoDocumento
- modificarParrafoDocumento
- obtenerParrafosDocumento

2.4.1.2.68 TrParrafoTipoDocumento

trewa.bd.trapi.trapiui.tpo.TrParrafoTipoDocumento

Clase que representa un párrafo para un tipo de documento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPARTIPODOC	TpoPK	Identificador del párrafo del tipo de documento.
→ PARRAFO	String	Texto del párrafo.
→ ORDEN	Integer	Orden del párrafo dentro del documento.
→ ALINEACION	String	Alineación del párrafo. → “ I ” – Izquierda → “ C ” – Centrado → “ D ” – Derecha → “ J ” – Justificado
→ ETIQUETA	String	Etiqueta del párrafo.
→ EDITABLE	String	Indica si el párrafo es editable. → “ S ” – Sí → “ N ” – No

→ ESTILO	String	Estilo del texto del párrafo. → " NORMAL " – Normal → " NEGRIT " – Negrita → " CURSIV " – Cursiva → " SUBRAY " – Subrayado → " NE-CU " – Negrita y Cursiva → " NE-SU " – Negrita y Subrayado → " CU-SU " – Cursiva y Subrayado → " NE-CU-SU " – Negrita, Cursiva y Subrayado
→ ESTILOETIQ	String	Estilo de la etiqueta del párrafo. → " NORMAL " – Normal → " NEGRIT " – Negrita → " CURSIV " – Cursiva → " SUBRAY " – Subrayado → " NE-CU " – Negrita y Cursiva → " NE-SU " – Negrita y Subrayado → " CU-SU " – Cursiva y Subrayado → " NE-CU-SU " – Negrita, Cursiva y Subrayado
→ TIPODOC	TrTipoDocumento	Tipo de documento.
→ TIPOPARR	TrTipoParrafo	Tipo de párrafo.
→ FUSIONAR	String	Indica si se fusionan o no variables al generar. → " S " – Sí → " N " – No
→ IMAGEN	byte[]	Imagen del párrafo.
→ FORMATO	String	Tipo de formato de la imagen (tipo mime).
→ NOMBREFICHERO	String	Nombre del fichero del documento físico.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPARRTIPO DOC	NUMÉRICO	Identificador del párrafo del tipo de documento.
→ CAMPO_PARRAFO	CADENA	Texto del párrafo
→ CAMPO_ORDEN	NUMÉRICO	Orden del párrafo dentro del documento.
→ CAMPO_ALINEACION	CADENA(1)	Alineación del párrafo. → " I " – Izquierda → " C " – Centrado → " D " – Derecha → " J " – Justificado
→ CAMPO_ETIQUETA	CADENA(15)	Etiqueta del párrafo.
→ CAMPO_EDITABLE	CADENA(1)	Indica si el párrafo es editable. → " S " – Sí → " N " – No
→ CAMPO_ESTILO	CADENA(8)	Estilo del texto del párrafo.

→ CAMPO_ESTILOETIQ	CADENA(8)	→ " NORMAL " – Normal → " NEGRIT " – Negrita → " CURSIV " – Cursiva → " SUBRAY " – Subrayado → " NE-CU " – Negrita y Cursiva → " NE-SU " – Negrita y Subrayado → " CU-SU " – Cursiva y Subrayado → " NE-CU-SU " – Negrita, Cursiva y Subrayado Estilo de la etiqueta del párrafo. → " NORMAL " – Normal → " NEGRIT " – Negrita → " CURSIV " – Cursiva → " SUBRAY " – Subrayado → " NE-CU " – Negrita y Cursiva → " NE-SU " – Negrita y Subrayado → " CU-SU " – Cursiva y Subrayado → " NE-CU-SU " – Negrita, Cursiva y Subrayado
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador el tipo de documento.
→ CAMPO_REFTIPOPARR	NUMÉRICO	Identificador del tipo de párrafo.
→ CAMPO_FUSIONAR	CADENA(1)	Indica si se fusionan o no variables al generar. → " S " – Sí → " N " – No
→ CAMPO_FORMATO	CADENA(128)	Tipo de formato de la imagen (tipo mime)
→ CAMPO_NOMBREFICHERO	CADENA(64)	Nombre del fichero del documento físico

Métodos para aplicación de los filtros

→ obtenerParrafoTipoDocumento

2.4.1.2.69 TrPerfilUsuario

trewa.bd.trapi.trapiui.tpo.TrPerfilUsuario

Clase que representa la información de los distintos perfiles de usuario que pueden existir.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPERFILUSU	TpoPK	Identificador del perfil de usuario
→ NOMBRE	String	Nombre del perfil de usuario

- **DESCRIPCION** String Descripción del perfil de usuario
- **STMA** TrSistema Sistema del perfil de usuario.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPERFILUSU	NUMÉRICO	Identificador del perfil de usuario
→ CAMPO_NOMBRE	CADENA(25)	Nombre del perfil de usuario
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del perfil de usuario
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema del perfil de usuario
→ CAMPO_CODSTMA	CADENA(10)	Código o nombre del sistema del perfil de usuario
→ CAMPO_DESCSTMA	CADENA(128)	Descripción del sistema del perfil de usuario

Métodos para aplicación de los filtros

- obtenerPerfiles
- obtenerPerfilesUsuario
- obtenerPerfilUsuario

2.4.1.2.70TrPeticiónEliminada

trewa.bd.trapi.trapiui.tpo.TrPeticiónEliminada

Clase que representa la información resultante de la eliminación de una o varias peticiones de firma con port@firmas.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ hashPetición	String	Código hash de la petición.

→ **eliminada** Boolean Indica si se ha eliminado la petición.
 → true, la petición se ha eliminado
 → false, la petición no se ha eliminado

Métodos para uso de la clase → eliminarPeticiónPortafirmas

2.4.1.2.71 TrPlantillaProcedimiento

trewa.bd.trapi.trapiui.tpo.TrPlantillaProcedimiento

Clase que representa las posibles plantillas del procedimiento necesarias para comenzar un expediente o que servirán a lo largo de la vida de su tramitación.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPLANTPROC	TpoPK	Identificador de la plantilla del procedimiento.
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento.
→ NOMBRE	String	Nombre de la plantilla del procedimiento.
→ DESCRIPCION	String	Descripción de la plantilla del procedimiento.
→ FICHERO	byte[]	Fichero de la plantilla del procedimiento.
→ NOMBREFICHERO	String	Nombre del fichero físico de la plantilla.
→ FORMATO	String	Tipo de formato del fichero (tipo mime).

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPLANTPROC	NUMÉRICO	Identificador de la plantilla del procedimiento.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_NOMBRE	CADENA(100)	Nombre de la plantilla del procedimiento.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la plantilla del procedimiento.
→ CAMPO_NOMBREFICHERO	CADENA(64)	Nombre del fichero físico de la plantilla.
→ CAMPO_FORMATO	CADENA(128)	Tipo de formato del fichero (tipo mime).

Métodos para aplicación de los filtros → obtenerPlantillaProcedimiento

2.4.1.2.72 TrPlazo

trewa.bd.trapi.trapiui.tpo.TrPlazo

Clase que representa la unidad de tiempo, el número de unidades y la descripción de la fecha en la que cumple el plazo.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ UNIDAD	String	Indica la unidad de tiempo → “ D ” – Días → “ M ” – Meses → “ A ” – Años
→ NUMUNIDADES	Integer	Número de unidades que se cuentan (nº de días, nº de años,...)
→ DESCFECHALIMIT E	String	Descripción de la fecha límite

Métodos para aplicación de los filtros

- obtenerBloquesPermitidos
- obtenerDatosTransicion
- obtenerDocumentosPermitidos
- obtenerTareasPermitidas
- recorrerFasesPosTipEv

2.4.1.2.73 TrProvincia

trewa.bd.trapi.trapiui.tpo.TrProvincia

Clase que representa la información de una provincia.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODPROVINCIA	String	Identificador de la provincia
→ NOMBRE	String	Nombre de la provincia.
→ DESCRIPCION	String	Descripción de la provincia.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODPROVINCIA	CADENA(2)	Identificador de la provincia
→ CAMPO_NOMBRE	CADENA(32)	Nombre de la provincia.
→ CAMPO_DESCRIPCION	CADENA(64)	Descripción de la provincia.

Métodos para aplicación de los filtros

- obtenerDatosContactoInteresado
- obtenerDatosContactoo
- obtenerProvincia

2.4.1.2.74 TrPtoTrabOrganismo

trewa.bd.trapi.trapiui.tpo.TrPtoTrabOrganismo

Clase que representa los distintos puestos de trabajo que existen en cada organismo.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ ORGANISMO	TrOrganismo	Organismo
→ PUESTOTRABAJO	TrPuestoTrabajo	Puesto de trabajo

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODPTOTRAB	CADENA(10)	Identificador del puesto de trabajo
→ CAMPO_NOMBREPTOTRAB	CADENA(32)	Nombre del puesto de trabajo
→ CAMPO_DESCRIPCIONPTOTRAB	CADENA(64)	Descripción del puesto de trabajo
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo
→ CAMPO_CODMUNICIPIO	CADENA(3)	Código del municipio
→ CAMPO_CODPROVINCIA	CADENA(2)	Código de la provincia
→ CAMPO_TIPOVIA	CADENA(10)	Tipo de vía
→ CAMPO_CODORG	CADENA(16)	Código del organismo
→ CAMPO_NOMBREORG	CADENA(32)	Nombre del organismo
→ CAMPO_DESCRIPCIONORG	CADENA(200)	Descripción del organismo
→ CAMPO_TIPO	CADENA(1)	Tipo de organismo → 'DP' – Delegación Provincial → 'SC' – Servicios Centrales → 'CE' – Centro Adscrito → 'EE' – Empresa Externa
→ CAMPO_NOMBREVIA	CADENA(64)	Nombre de la vía
→ CAMPO_NUMERO	NUMÉRICO	Número del domicilio
→ CAMPO_LETRA	CADENA(2)	Letra del domicilio
→ CAMPO_ESCALERA	CADENA(2)	Escalera del domicilio
→ CAMPO_PISO	NUMERICO	Piso del domicilio



→	CAMPO_PUERTA	CADENA(2)	Puerta del domicilio
→	CAMPO_CODPOSTAL	NUMERICO	Código postal del domicilio
→	CAMPO_TELEFONO	CADENA(25)	Número(s) de teléfono de contacto
→	CAMPO_TLFMOVIL	CADENA(25)	Número(s) de teléfono móvil
→	CAMPO_FAX	CADENA(25)	Número(s) de Fax
→	CAMPO_EMAIL	CADENA(64)	Correo electrónico
→	CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA
→	CAMPO_IDARIES	CADENA(5)	Identificador de aries
→	CAMPO_IDENTIFICADOR	CADENA(15)	Número de identificador
→	CAMPO_DIGCONTROL	CADENA(1)	Dígito de control
→	CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia del organismo
→	CAMPO_FECHAFINIVIG	DATE	Fecha de fin de vigencia del organismo
→	CAMPO_REFORGPADRE	NUMÉRICO	Identificador del organismo superior al que pertenece.
→	CAMPO_REFTIPOORG	NUMÉRICO	Identificador del tipo de organismo.
→	CAMPO_REFTIPOORGZ	NUMÉRICO	Identificador del tipo de organización.
→	CAMPO_ORDENPRELACION	CADENA(15)	Orden de relación del organismo

Métodos para aplicación de los filtros

→ obtenerPtoTrabOrganismo

2.4.1.2.75 TrPuestoTrabajo

trewa.bd.trapi.trapiui.tpo.TrPuestoTrabajo

Clase que representa los distintos puestos de trabajo.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODPTOTRAB	String	Identificador o código del puesto de trabajo
→ NOMBRE	String	Nombre del puesto de trabajo
→ DESCRIPCION	String	Descripción del puesto de trabajo

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODPTOTRAB	CADENA(10)	Identificador del puesto de trabajo
→ CAMPO_NOMBRE	CADENA(32)	Nombre del puesto de trabajo
→ CAMPO_DESCRIPCION	CADENA(64)	Descripción del puesto de trabajo

→ obtenerEmpleado

Métodos para aplicación de los filtros

→ obtenerPtoTrabOrganismo

→ obtenerPuestoTrabajo

2.4.1.2.76 TrRazonInteres

trewa.bd.trapi.trapiui.tpo.TrRazonInteres

Clase que recoge las distintas razones de interés de los interesados en los expedientes y documentos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFRAZONINT	TpoPK	Identificador de la razón de interés
→ ABREVIATURA	String	Abreviatura de la razón de interés
→ DESCRIPCION	String	Descripción de la razón de interés
→ OBSOLETO	String	Indica si la razón de interés está obsoleta. → ' S ' – Está obsoleta → ' N ' – No está obsoleta
→ CODWANDA	String	Valor en w@ndA de la razón de interés

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFRAZONINT	NUMÉRICO	Identificador de la razón de interés
→ CAMPO_ABREVIATURA	CADENA(10)	Abreviatura de la razón de interés
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción de la razón de interés
→ CAMPO_OBSOLETO	CADENA(1)	Indica si la razón de interés está obsoleta. → ' S ' – Está obsoleta → ' N ' – No está obsoleta
→ CAMPO_CODWANDA	CADENA(10)	Valor en w@ndA de la razón de interés

Métodos para aplicación de los filtros

- obtenerInteresadoDocumento
- obtenerInteresadoExpediente
- obtenerInteresadoExpedienteCC
- obtenerInteresadoExpedienteDadoDeBaja
- obtenerRazonesInteres

2.4.1.2.77TrRazonInteresDocumento

trewa.bd.trapi.trapiui.tpo.TrRazonInteresDocumento

Clase que recoge las distintas razones de interés asociada a los documentos permitidos dependientes de un procedimiento, para indicar las posibles asociaciones de interesados a los documentos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDOCPERM	TpoPK	Identificador del documento permitido al que está asociada la razón de interés de procedimiento
→ RAZONINTPROC	TrRazonInteresProcedimiento	Razón de interés de procedimiento
→ CARDINALIDAD_MAX	Integer	Número máximo requerido de interesados al documento
→ CARDINALIDAD_MIN	Integer	Número mínimo requerido de interesados al documento
→ N_FIRMA_MAX	Integer	Número máximo de firmas requerido para el documento por el interesado
→ N_FIRMA_MIN	Integer	Número mínimo de firmas requerido para el documento por el interesado
→ EDITAR	String	Indica si el interesado puede realizar acciones sobre el documento. → ' S ' – Puede → ' N ' – No puede
→ TELEMATICA	String	Indica si el interesado puede presentar telemáticamente el documento. → ' S ' – Presentación telemática → ' N ' – No presentación telemática

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPROC	NUMÉRICO	Identificador del procedimiento
→ CAMPO_REFRAZONINTPROC	NUMÉRICO	Identificador de la razón de interés
→ CAMPO_ABREVRACIONINT	CADENA(100)	Abreviatura de la razón de interés
→ CAMPO_DESCRAZONINT	CADENA(100)	Descripción de la razón de interés
→ CAMPO_CARDINALIDAD_MAX	NUMÉRICO	Cardinalidad máxima de la razón interés asociada
→ CAMPO_CARDINALIDAD_MIN	NUMÉRICO	Cardinalidad mínima de la razón interés asociada
→ CAMPO_N_FIRMA_MAX	NUMÉRICO	Número de firmas máxima requeridas para el documento por el interesado
→ CAMPO_N_FIRMA_MIN	NUMÉRICO	Número de firmas mínimo requeridas para el documento por el interesado
→ CAMPO_EDITAR	CADENA(1)	Indica si el interesado puede realizar acciones sobre el documento. → ' S ' – Puede

- **CAMPO_TELEMATICA** CADENA(1) → '**N**' – No puede
Indica si el interesado puede presentar telemáticamente el documento.
- **CAMPO_REFDOCPER** NUMÉRICO → '**S**' – Presentación telemática
→ '**N**' – No presentación telemática
Identificador del documento permitido
- **Métodos para aplicación de los filtros** → obtenerRazonInteresDocumento

2.4.1.2.78 TrRazonInteresProcedimiento

trewa.bd.trapi.trapiui.tpo.TrRazonInteresProcedimiento

Clase que recoge las distintas razones de interés asociada a los procedimientos, para indicar las posibles asociaciones de interesados a los expedientes.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDEFPROC	TpoPK	Identificador del procedimiento
→ RAZONINT	TrRazonInteres	Razón de interés
→ CARDINALIDAD_M AX	Integer	Número máximo requerido de interesados al expediente
→ CARDINALIDAD_M IN	Integer	Número mínimo requerido de interesados al expediente
→ COMENTARIO	String	Comentarios sobre la razón de interés
→ OBLIG_REL_APP_E LEC	String	Indica si el interesado se encuentra obligado a relacionarse con las aplicaciones por medios electrónicos. Sus valores pueden ser: → ' S ' – Sí → ' N ' – No, por defecto, se toma el valor de 'N'.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador del procedimiento
→ CAMPO_REFRAZONINT	NUMÉRICO	Identificador de la razón de interés
→ CAMPO_ABREVRazonINT	CADENA(10)	Abreviatura de la razón de interés
→ CAMPO_DESCRAZONINT	CADENA(100)	Descripción de la razón de interés
→ CAMPO_CARDINALIDAD_M AX	NUMÉRICO	Cardinalidad máxima de la razón interés asociada
→ CAMPO_CARDINALIDAD_M IN	NUMÉRICO	Cardinalidad mínima de la razón interés asociada
→ CAMPO_OBSOLETO_RI	CADENA(1)	Indica si la razón de interés está obsoleta. → ' S ' – Está obsoleta → ' N ' – No está obsoleta
→ CAMPO_COMENTARIO	CADENA	Comentarios sobre la razón de interés
→ CAMPO_OBLIG_REL_APP_E	CADENA(1)	Indica si el interesado se encuentra obligado a relacionarse con las aplicaciones por medios electrónicos. Sus valores pueden

LEC

ser:

→ 'S' – Está obsoleta

→ 'N' – No, por defecto, se toma el valor "N"

Métodos para aplicación de los filtros

→ obtenerRazonInteresProcedimiento

2.4.1.2.79 TrRegistroOperacion

trewa.bd.trapi.trapiui.tpo.TrRegistroOperacion

Clase que representa un registro de una operación realizada sobre una relación entre interesados o un límite de éstas relaciones. Normalmente se utiliza para crear un log o histórico de las operaciones que se han llevado a cabo de representación, apoderamientos, etc.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFREGOPERACION	TpoPK	Identificador del registro de la operación
→ REFRELACIONINTE	TpoPK	Identificador de la relación entre interesados
→ REFLIMITERELA	TpoPK	Identificador del límite de la relación. Opcional
→ FECHA	Timestamp	Fecha de la operación
→ DESCRIPCION	String	Descripción

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFREGOPERACION	NUMÉRICO	Identificador del registro de la operación
→ CAMPO_REFRELACIONINTE	NUMÉRICO	Identificador de la relación entre interesados
→ CAMPO_REFLIMITERELA	NUMÉRICO	Identificador del límite de la relación. Opcional
→ CAMPO_FECHA	DATE	Fecha de la operación
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción

Métodos para aplicación de los filtros

obtenerRegistroOperacion

2.4.1.2.80 TrRelacionDefinida

trewa.bd.trapi.trapiui.tpo.TrRelacionDefinida

Clase que representa la información referente a relaciones establecidas entre fases, transiciones, definición de procedimientos,... definidas en el tramitador.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ TIPORELACION	TrTipoRelacion	Tipo de relación
→ REFRELACION	TpoPK	Identificador de la relación
→ DESCRIPCION	String	Descripción de la relación
→ REFFASEA	TpoPK	Identificador de la fase A
→ REFFASEB	TpoPK	Identificador de la fase B
→ TRANSICION	TrTransicion	Transición
→ TIPODOC	TrTipoDocumento	Tipo de documento
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPORELA	NUMÉRICO	Identificador del tipo de relación
→ CAMPO_NOMBRETIPORELA	CADENA(25)	Nombre del tipo de relación
→ CAMPO_DESCRIPCIONTIPORELA	CADENA(250)	Descripción del tipo de relación
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
→ CAMPO_REFRELACION	NUMÉRICO	Identificador de la relación
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la relación
→ CAMPO_REFFASEA	NUMÉRICO	Identificador de la fase A
→ CAMPO_REFFASEB	NUMÉRICO	Identificador de la fase B
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento

Métodos para aplicación de los filtros

→ obtenerRelacionesDefinidas

2.4.1.2.81 TrRelacionExpediente

trewa.bd.trapi.trapiui.tpo.TrRelacionExpediente

Clase que recoge las posibles relaciones entre los distintos expedientes del sistema para dar soporte al anidamiento, encadenamiento o simple relación entre expedientes..

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFRELACIONEXP	TpoPK	Identificador de la relación entre expedientes.
→ REFEXPEDIENTE	TpoPK	Identificador del expediente (interno o externo).
→ TIPORELACION	String	Tipo de relación → 'M' – Maestro → 'D' – Detalle → 'I' – Igualdad
→ OBSERVACIONES	String	Observaciones a la relación.
→ COMPONENTE	TrComponente	Componente
→ NUMEXP	String	Número del expediente en la aplicación "cliente".
→ TITULOEXP	String	Título del expediente en la aplicación cliente.
→ PROPIETARIOEXP	String	Usuario propietario del expediente
→ IDENTIFICADORE NI	String	Identificador ENI del expediente

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFRELACIONEXP	NUMÉRICO	Identificador de la relación entre expedientes.
→ CAMPO_REFEXPEDIENTE	NUMÉRICO	Identificador del expediente (interno o externo).
→ CAMPO_TIPORELACION	CADENA(1)	Tipo de relación → 'M' – Maestro → 'D' – Detalle → 'I' – Igualdad
→ CAMPO_OBSERVACIONES	CADENA(250)	Observaciones a la relación.
→ CAMPO_REFCOMP	NUMÉRICO	Identificador del componente
→ CAMPO_COMPONENTE	CADENA(50)	Nombre del componente
→ CAMPO_NUMEXP	CADENA(30)	Número del expediente en la aplicación "cliente".
→ CAMPO_TITULOEXP	CADENA(250)	Título del expediente en la aplicación cliente.
→ CAMPO_PROPIETARIOEXP	CADENA(10)	Usuario propietario del expediente
→ CAMPO_ENI	CADENA(55)	Identificador normalizado ENI del expediente
→ CAMPO_REFEXPEXEXPE	NUMÉRICO	Identificador del expediente con que se relaciona

Métodos para aplicación de los filtros y dónde se usa

- altaExpediente
- obtenerRelacionExpediente
- relacionarExpedientePorENI

2.4.1.2.82 TrRelacionInteresado

trewa.bd.trapi.trapiui.tpo.TrRelacionInteresado

Clase que recoge las posibles relaciones entre los ciudadanos interesados.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ INTERESADO	TrInteresado	Interesado
→ TIPOCONTACTO	TrTipoContacto	Tipo de contacto
→ SENTIDORELACION	String	Sentido de la relación
→ REFINTERESADOINI	TpoPK	Identificador del interesado inicio
→ REFRELACIONINTE	TpoPK	Identificador de la relación entre interesados
→ FECHAINIVIG	Timestamp	Fecha de inicio de vigencia de la relación
→ FECHAFINIVIG	Timestamp	Fecha de fin de vigencia de la relación
→ REPRESENTA	String	Indica si la relación es de representación. Sus valores pueden ser "S" o "N"
→ PENDIENTE	String	Indica si está pendiente. Sus valores pueden ser "S" ó "N".
→ REFDEFPROC	TpoPK	Identificador del procedimiento
→ FECHAREVOCACION	Timestamp	Fecha de revocación
→ FECHARECHAZO	Timestamp	Fecha de rechazo

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOCONT	NUMÉRICO	Identificador del tipo de contacto
→ CAMPO_ABREVTIPOCONTACTO	CADENA(10)	Abreviatura del tipo de contacto
→ CAMPO_REFINTERESADO	NUMÉRICO	Identificador del interesado
→ CAMPO_CODTIPOIDENT	CADENA(10)	Tipo de identificación
→ CAMPO_NUMIDENT	CADENA(15)	Número de identificación.
→ CAMPO_APELLIDO1	CADENA(50)	Primer apellido del interesado.
→ CAMPO_APELLIDO2	CADENA(50)	Segundo apellido del interesado.
→ CAMPO_NOMBRE	CADENA(100)	Nombre del interesado.
→ CAMPO_SEXO	CADENA(1)	Sexo del interesado. → 'F' – Femenino → 'M' – Masculino
→ CAMPO_DIGCONTROL	CADENA(1)	Dígito de control para el número de identificación.
→ CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA.
→ CAMPO_FECHANACIM	DATE	Fecha de nacimiento.
→ CAMPO_FECHABAJA	DATE	Fecha de baja en el sistema.
→ CAMPO_NUMREGMER	CADENA(30)	Número del registro mercantil.
→ CAMPO_FECHARCA	DATE	Fecha de RCA de la entidad.



→	CAMPO_EPIGRAFEIAE	CADENA(30)	Epígrafe IAE.
→	CAMPO_EPIGRAFECNAE	CADENA(30)	Epígrafe CNAE.
→	CAMPO_COMENTARIOS	CADENA(100)	Comentarios.
→	CAMPO_REFTIPOORGZ	CADENA(1)	Identificador del tipo de organización.
→	CAMPO_ANAGRAMAFISCAL	CADENA(20)	Anagrama fiscal del interesado
→	CAMPO_REFDATOCONT	NUMÉRICO	Identificador de los datos de contacto
→	CAMPO_REFRELACIONINTE	NUMÉRICO	Identificador de la relación entre interesados
→	CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia de la relación
→	CAMPO_FECHAFINVIG	DATE	Fecha de fin de vigencia de la relación
→	CAMPO_REPRESENTA	CADENA(1)	Indica si la relación es de representación. Sus valores pueden ser "S" o "N"
→	CAMPO_PENDIENTE	CADENA(1)	Indica si está pendiente. Sus valores pueden ser "S" ó "N".
→	CAMPO_REFDEFPROC	NUMÉRICO	Identificador del procedimiento
→	CAMPO_FECHAREVOCA	DATE	Fecha de revocación
→	CAMPO_FECHARECHAZO	DATE	Fecha de rechazo
→	CAMPO_SENTIDORELACION	CADENA(2)	Sentido de la relación. Sus valores pueden ser "AB" cuando el interesado que se consulta es A ó "BA" cuando el interesado que se consulta es B.

Métodos para aplicación de los filtros y dónde se usa

- insertarRelacionInteresado
- modificarInteresado
- obtenerRelacionesInteresado

2.4.1.2.83TrSistema

trewa.bd.trapi.trapiui.tpo.TrSistema

Clase que representa la información referente a un sistema.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	AUDITORIA	TrAuditoria	Datos de auditoría
→	REFSTMA	TpoPK	Identificador del sistema
→	CODSTMA	String	Código del sistema
→	DESCRIPCION	String	Descripción del sistema
→	COMPONENTEAR CH	TrComponente	Componente de archiva
→	COMPONENTEAVI	TrComponente	Componente del avisador

- **COMPONENTEFIRMA** TrComponente Componente que firma los documentos
- **COMPONENTEGUAR** TrComponente Componente que guarda los documentos
- **COMPONENTEHC** TrComponente Componente para configurar repositorios externos
- **COMPONENTEINF** TrComponente Componente a informar
- **COMPONENTENO** TrComponente Componente que notifica los documentos
- **COMPONENTERE** TrComponente Componente que registra los documentos
- **CONEXIONBUS** String Indica si se tiene conexión al bus. Sus valores pueden ser:
→ 'S' – Sí
→ 'N' – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
→ CAMPO_CODSTMA	CADENA(10)	Código del sistema
→ CAMPO_DESCRIPCION	CADENA(128)	Descripción del sistema
→ CAMPO_CONEXIONBUS	CADENA(1)	Indica si se tiene conexión al bus. → 'S' – Sí → 'N' – No → 'O' – Opcional
→ CAMPO_REFCOMPINF	NUMÉRICO	Identificador del componente a informar.
→ CAMPO_NOMBRECOMPINF	CADENA(50)	Nombre del componente que informa al bus
→ CAMPO_REFCOMPAVI	NUMÉRICO	Identificador del componente del @visorador.
→ CAMPO_NOMBRECOMPAVI	CADENA(50)	Nombre del componente @visorador
→ CAMPO_REFCOMPGUAR	NUMÉRICO	Identificador del componente que guarda los documentos.
→ CAMPO_NOMBRECOMPGUAR	CADENA(50)	Nombre del componente w@rdA
→ CAMPO_REFCOMPARCH	NUMÉRICO	Identificador del componente que archiva.
→ CAMPO_NOMBRECOMPARCH	CADENA(50)	Nombre del componente que archiva
→ CAMPO_REFCOMPREG	NUMÉRICO	Identificador del componente que registra los documentos.
→ CAMPO_NOMBRECOMPREG	CADENA(50)	Nombre del componente de registro
→ CAMPO_REFCOMPFIRMA	NUMÉRICO	Identificador del componente que firma los documentos.
→ CAMPO_NOMBRECOMPFIRMA	CADENA(50)	Nombre del componente de firma
→ CAMPO_REFCOMPNOTI	NUMÉRICO	Identificador del componente que notifica los documentos.
→ CAMPO_NOMBRECOMPNOTI	CADENA(50)	Nombre del componente de notificaciones
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICA	DATE	Fecha de modificación del registro

- **CAMPO_NOMBRECOMPFCV** CADENA Nombre del componente HCV
- **CAMPO_REFCOMPFCV** NUMÉRICO Identificador del componente de HCV

- explorarExpedientesSimple
- obtenerAccionesDocumento
- obtenerAccionTransicion
- obtenerBloquesDefinidos
- obtenerBloquesPermitidos
- obtenerCondicionesDocumento
- obtenerCondicionTransicion
- obtenerConstante
- obtenerDatosFase
- obtenerDefProcedimientosDefinidos
- obtenerDocumentosExpediente
- obtenerDocumentosExpedienteSimple
- obtenerDocumentosPermitidos
- obtenerExpedientes
- obtenerExpedientesCaducados
- obtenerExpsRelacionInteresado
- obtenerFases
- obtenerFasesFinTransicion
- obtenerFasesProcedimiento
- obtenerInteresadoExpedienteCC
- obtenerMetafases
- obtenerMetafasesProcedimiento
- obtenerParametrosBloques
- obtenerPerfiles
- obtenerPerfilesUsuario
- obtenerPerfilUsuario
- obtenerProcedimientosUsuario
- obtenerRelacionesDefinidas
- obtenerSistema
- obtenerTipoDocumentoSimple
- obtenerTipoExpedienteSimple
- obtenerTiposActo
- obtenerTiposActoAdmExpediente
- obtenerTiposDocumento
- obtenerTiposParrafo
- obtenerVariablesSistema
- obtenerVersionDefProcedimiento
- recorreFasesPosTipEv

Métodos para aplicación de los filtros y dónde se usa

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.1.2.84 TrTareaExpediente

trewa.bd.trapi.trapiui.tpo.TrTareaExpediente

Clase que representa una tarea del expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTAREAEXP	TpoPK	Identificador de la tarea del expediente
→ REFTAREA	TpoPK	Identificador de la tarea
→ NOMBRETAREA	String	Nombre de la tarea
→ TIPO	String	Tipo de la tarea → “ G ” – Generar → “ I ” – Incorporar → “ M ” – Manipular datos → “ O ” – Otras
→ FECHACOMIENZO	Timestamp	Fecha de comienzo de la tarea
→ FECHALIMITE	Timestamp	Fecha límite de realización de la tarea
→ FECHAFINAL	Timestamp	Fecha de finalización de la tarea
→ ESTADO	String	Estado de la tarea. Si la tarea es de tipo M ó O → “ I ” – Iniciada → “ D ” – Descartada → “ F ” – Finalizada Si la tarea es de tipo G ó I → “ R ” – El documento se encuentra en realización → “ E ” – El documento se encuentra pendiente de firma → “ F ” – El documento se encuentra firmado → “ D ” – El documento está descartado



→	OBSERVACIONES	String	→ "T" – El documento está terminado
→	REFFASE	TpoPK	→ "V" – El documento está versionado
→	NOMBREFASE	String	Observaciones de la tarea
→	REFEXPPHASE	TpoPK	Identificador de la fase
→	REFDEFPROC	TpoPK	Nombre de la fase
→	USUARIO	String	Identificador del expediente en fase
→	NOMBREUSU	String	Identificador de la definición del procedimiento
→	MULTIPLE	String	Usuario
			Nombre y apellidos del usuario
			Indica si es múltiple
			→ "S" – Sí
			→ "N" – No
→	TEXTOAUXILIAR	String	Texto auxiliar
→	REFTAREALLAMANTE	TpoPK	Identificador de la tarea llamante
→	NOMBRETAREALLAMANTE	String	Nombre de la tarea llamante
→	INFORMADABUS	String	Indica si se informa al bus del inicio de la tarea
			→ "S" – Sí
			→ "N" – No
→	REFEXPEDIENTE	TpoPK	Identificador del expediente
→	NUMEXP	String	Número del expediente
→	TITULOEXP	String	Título del expediente
→	OBSERVACIONESEXP	String	Observaciones del expediente
→	PROPIETARIOEXP	String	Usuario propietario del expediente
→	REFORGEXP	TpoPK	Identificador del organismo al que pertenece el expediente
→	REFORGENVEXP	TpoPK	Identificador del organismo que envía el expediente
→	REFSTMA	TpoPK	Identificador del sistema de la tarea.
→	CODSTMA	String	Código del sistema de la tarea.
→	DESCSTMA	String	Descripción del sistema de la tarea.
→	DESCFASE	String	Descripción de la fase
→	DESCTAREA	String	Descripción de la tarea
→	DESCTAREALLAMANTE	String	Descripción de la tarea llamante

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTAREAEXP	NUMÉRICO	Identificador de la tarea del expediente
→ CAMPO_REFTAREA	NUMÉRICO	Identificador de la tarea
→ CAMPO_NOMBRETAREA	CADENA(50)	Nombre de la tarea
→ CAMPO_TIPO	CADENA(1)	Tipo de la tarea
		→ "G" – Generar
		→ "I" – Incorporar
		→ "M" – Manipular datos
		→ "O" – Otras

→	CAMPO_FECHACOMIENZO	DATE	Fecha de comienzo de la tarea
→	CAMPO_FECHALIMITE	DATE	Fecha límite de realización de la tarea
→	CAMPO_FECHAFINAL	DATE	Fecha de finalización de la tarea
→	CAMPO_ESTADO	CADENA(1)	Estado de la tarea → "I" – Iniciada → "D" – Descartada → "F" – Finalizada
→	CAMPO_OBSERVACIONES	CADENA(250)	Observaciones de la tarea
→	CAMPO_REFFASE	NUMÉRICO	Identificador de la fase
→	CAMPO_NOMBREFASE	CADENA(50)	Nombre de la fase
→	CAMPO_REFEXPFASE	NUMÉRICO	Identificador del expediente en fase
→	CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→	CAMPO_USUARIO	CADENA(10)	Usuario
→	CAMPO_NOMBREUSU	CADENA(98)	Nombre y apellidos del usuario
→	CAMPO_MULTIPLE	CADENA(1)	Indica si es múltiple → "S" – Sí → "N" – No
→	CAMPO_TEXTOAUXILIAR	CADENA(50)	Texto auxiliar
→	CAMPO_REFTAREALLAMANTE	NUMÉRICO	Identificador de la tarea llamante
→	CAMPO_NOMBRETAREALLAMANTE	CADENA(50)	Nombre de la tarea llamante
→	CAMPO_INFORMADABUS	CADENA(1)	Indica si se informa al bus del inicio de la tarea → "S" – Sí → "N" – No
→	CAMPO_NUMEXP	CADENA(30)	Número del expediente
→	CAMPO_TITULOEXP	CADENA(250)	Título del expediente
→	CAMPO_OBSERVACIONESEXP	CADENA(250)	Observaciones del expediente
→	CAMPO_PROPIETARIOEXP	CADENA(10)	Usuario propietario del expediente
→	CAMPO_REFORGENVEXP	NUMÉRICO	Identificador del organismo al que pertenece el expediente
→	CAMPO_REFORGENVEXP	NUMÉRICO	Identificador del organismo que envía el expediente
→	CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema de la tarea.
→	CAMPO_CODSTMA	CADENA(10)	Código del sistema de la tarea.
→	CAMPO_DESCSTMA	CADENA(128)	Descripción del sistema de la tarea.
→	CAMPO_DESCFASE	CADENA	Descripción de la fase
→	CAMPO_DESCTAREA	CADENA	Descripción de la tarea
→	CAMPO_DESCTAREALLAMANTE	CADENA	Descripción de la tarea llamante

Métodos para aplicación de los filtros

→ obtenerTareasExpediente

2.4.1.2.85 TrTareaPermitida

trewa.bd.trapi.trapiui.tpo.TrTareaPermitida

Clase que representa la información de una tarea permitida.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTAREAFASE	TpoPK	Identificador de la tarea en fase
→ REFTAREA	TpoPK	Identificador de la tarea
→ TAREA	String	Nombre de la tarea
→ ETIQUETA	String	Etiqueta de la tarea
→ ETIQUETALARGA	String	Etiqueta larga para la tarea
→ DESCRIPCION	String	Descripción de la tarea
→ TIPO	String	Tipo de la tarea permitida → “ G ” – Generar → “ I ” – Incorporar → “ M ” – Manipular datos → “ O ” – Otras
→ PERMISO	String	Permiso de la tarea
→ OBLIGATORIA	String	Indica si es obligatoria su realización → “ S ” – Sí → “ N ” – No
→ PLAZO	TrPlazo	Plazo de realización
→ MULTIPLE	String	Indica si es múltiple → “ S ” – Sí → “ N ” – No
→ TEXTOAUXILIAR	String	Texto auxiliar
→ REFTAREALLAMANTE	TpoPK	Identificador de la tarea llamante
→ TAREALLAMANTE	String	Nombre de la tarea llamante
→ ORDEN	Integer	Orden de la tarea
→ INFORMARBUS	String	Indica si se informa al bus de la realización de la tarea → “ S ” – Sí → “ N ” – No
→ REFTIPOACTO	TpoPK	Identificador del tipo de acto
→ TEXTOAUXILIARTF	String	Texto auxiliar de la tarea en fase (documento permitido)

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTAREAFASE	NUMÉRICO	Identificador de la tarea en fase
→ CAMPO_REFTAREA	NUMÉRICO	Identificador de la tarea
→ CAMPO_TAREA	CADENA(50)	Nombre de la tarea
→ CAMPO_ETIQUETA	CADENA(11)	Etiqueta de la tarea

→	CAMPO_ETIQUETALARGA	CADENA(25)	Etiqueta larga para la tarea
→	CAMPO_DESCRIPCION	CADENA(50)	Descripción de la tarea
→	CAMPO_TIPO	CADENA(1)	Tipo de la tarea permitida → “ G ” – Generar → “ I ” – Incorporar → “ M ” – Manipular datos → “ O ” – Otras
→	CAMPO_PERMISO	CADENA(1)	Permiso de la tarea
→	CAMPO_OBLIGATORIA	CADENA(1)	Indica si es obligatoria su realización → “ S ” – Sí → “ N ” – No
→	CAMPO_MULTIPLE	CADENA(1)	Indica si es múltiple → “ S ” – Sí → “ N ” – No
→	CAMPO_TEXTOAUXILIAR	CADENA(50)	Texto auxiliar
→	CAMPO_REFTAREALLAMANTE	NUMÉRICO	Identificador de la tarea llamante
→	CAMPO_TAREALLAMANTE	CADENA(50)	Nombre de la tarea llamante
→	CAMPO_ORDEN	NUMÉRICO	Orden de la tarea
→	CAMPO_INFORMARBUS	CADENA(1)	Indica si se informa al bus de la realización de la tarea → “ S ” – Sí → “ N ” – No
→	CAMPO_DESCPLAZO	CADENA(100)	Descripción de la fecha límite del plazo
→	CAMPO_UNIDAD	CADENA(1)	Unidad de tiempo → “ D ” – Días → “ M ” – Meses → “ A ” – Años
→	CAMPO_NUMUNIDADES	NUMÉRICO	Número de unidades
→	CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto
→	CAMPO_TEXTOAUXILIARTF	CADENA	Texto auxiliar de la tarea en fase (documento permitido)

Métodos para aplicación de los filtros

→ obtenerTareasPermitidas

2.4.1.2.86 TrTextoDisposicion

trewa.bd.trapi.trapiui.tpo.TrTextoDisposicion

Clase que representa la información referente al texto de una disposición.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	REFDISPOSICION	TpoPK	Identificador del texto de la disposición
→	DESCRIPCION	String	Texto de la disposición

→ **TIPO** String Tipo de disposición
 → “**D**” – Decreto
 → “**T**” – Otros
 → “**O**” – Orden

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDISPOSICION	NUMÉRICO	Identificador del texto de la disposición
→ CAMPO_DESCRIPCION	CADENA(100)	Texto de la disposición
→ CAMPO_TIPO	CADENA(1)	Tipo de disposición → “ D ” – Decreto → “ T ” – Otros → “ O ” – Orden

Métodos para aplicación de los filtros

→ obtenerDisposicionesFirmante
 → obtenerFirmantesDefinidos
 → obtenerFirmantesFirmaDinamica
 → obtenerFirmantesTipoDocumento

2.4.1.2.87 TrTipoActo

trewa.bd.trapi.trapiui.tpo.TrTipoActo

Clase que representa a la información asociada a un tipo de acto administrativo.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOACTO	TpoPK	Identificador del acto administrativo
→ ABREVIATURA	String	Abreviatura del acto
→ DESCRIPCION	String	Descripción del acto administrativo
→ STMA	TrSistema	Sistema
→ TIPO	CADENA	Indica si el tipo de acto está asociado a una transición, documento o tarea. Sus valores pueden ser: → “ T ” – Transición → “ D ” – Documento → “ B ” – Bloque o tarea
→ TIPO_ACTO_BLOQUE	String	Tipo de acto para tarea bloque
→ TIPO_ACTO_DOCUMENTO	String	Tipo de acto para documento

→ **TIPO_ACTO_TRANSICION** String Tipo de acto para transición

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del acto administrativo
→ CAMPO_ABREVIATURA	CADENA(10)	Abreviatura del acto
→ CAMPO_DESCRIPCION	CADENA(200)	Descripción del acto administrativo
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema del tipo de acto
→ CAMPO_TIPO	CADENA()	Indica si el tipo de acto está asociado a una transición, documento o tarea. Sus valores pueden ser: → "T" – Transición → "D" – Documento → "B" – Bloque o tarea

→ obtenerAvisosCaducidadExpediente
→ obtenerBloquesPermitidos
→ obtenerDatosTransicion
→ obtenerDocumentosPermitidos
→ obtenerFasesExpediente
→ obtenerTiposActo
→ obtenerVariablesSistema
→ recorrerFasesPosTipEv
→ recorrerEvolExpte

Métodos para aplicación de los filtros

2.4.1.2.88 TrTipoCertificado

trewa.bd.trapi.trapiui.tpo.TrTipoCertificado

Recoge los distintos tipos de certificado que pueden existir asociados a un tipo de documento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOCERT	TpoPK	Identificador del tipo de certificado.
→ NOMBRE	String	Nombre del tipo de certificado.
→ DESCRIPCION	String	Descripción del tipo de certificado.
→ URL	String	URL del tipo de certificado
→ CLASE	String	Clase del tipo de certificado

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOCERT	NUMÉRICO	Identificador del tipo de certificado.
→ CAMPO_NOMBRE	CADENA(50)	Nombre del tipo de certificado.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del tipo de certificado.
→ CAMPO_URL	CADENA(255)	URL del tipo de certificado
→ CAMPO_CLASE	CADENA(250)	Clase del tipo de certificado

Métodos para aplicación de los filtros

- obtenerCertificados
- obtenerTiposCertificados
- obtenerTiposCertificadosProcedimiento

2.4.1.2.89 TrTipoCertificadoProcedimiento

trewa.bd.trapi.trapiui.tpo.TrTipoCertificadoProcedimiento

Recoge los distintos tipos de certificado asociado a procedimientos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDEFPROC	TpoPK	Identificador del procedimiento.
→ TIPOCERT	TrTipoCertificado	Tipo de certificado.
→ CODIGO_PETICION	String	Código petición SCSP.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador del tipo de certificado.
→ CAMPO_REFTIPCERT	CADENA(50)	Identificador del tipo de certificado.
→ CAMPO_NOMBRETIPCERT	CADENA(250)	Nombre del tipo de certificado.
→ CAMPO_DESCTIPCERT	CADENA(255)	Descripción del tipo de certificado.
→ CAMPO_URLTIPCERT	CADENA(250)	URL del tipo de certificado.
→ CAMPO_CLASEIMPLTIPCERT	CADENA(250)	Clase del tipo de certificado.
T		
→ CAMPO_CODPETICTIPCERT	CADENA(250)	Código petición SCSP.

Métodos para aplicación de los filtros

- obtenerTiposCertificadosProcedimiento

2.4.1.2.90 TrTipoContacto

trewa.bd.trapi.trapiui.tpo.TrTipoContacto

Recoge los distintos tipos de contacto que pueden existir entre ciudadanos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOCONT	TpoPK	Identificador del tipo de contacto.
→ ABREVIATURA	String	Abreviatura del tipo de contacto.
→ DESCRIPCION	String	Descripción del tipo de contacto.
→ CODWANDA	String	Valor en w@ndA del tipo de contacto.
→ OBSOLETO	String	Indica si el tipo de contacto está obsoleto. → “ S ” – Sí → “ N ” – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOCONT	NUMÉRICO	Identificador del tipo de contacto.
→ CAMPO_ABREVIATURA	CADENA(10)	Abreviatura del tipo de contacto.
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción del tipo de contacto.
→ CAMPO_CODWANDA	NUMÉRICO	Valor en w@ndA del tipo de contacto.
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de contacto está obsoleto. → “ S ” – Sí → “ N ” – No

Métodos para aplicación de los filtros

- obtenerRelacionesInteresado
- obtenerTipoContacto

2.4.1.2.91 TrTipoDocumento

trewa.bd.trapi.trapiui.tpo.TrTipoDocumento

Clase que representa la información referente a tipo de documento.

Atributos accesibles mediante métodos get/set



<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPODOC	TpoPK	Identificador del tipo de documento
→ ETIQUETA	String	Etiqueta del tipo de documento
→ NOMBRE	String	Nombre del tipo de documento
→ DESCRIPCION	String	Descripción del tipo de documento
→ ENTRADASALIDA	String	Indica cómo está definido el documento en TREW@ → “ E ” – Entrada → “ S ” – Salida → “ ES ” – Entrada / Salida
→ INCGEN	String	Indica cómo está definido el documento → “ I ” – Documento a Incorporar al expediente → “ G ” – Documento a Generar
→ STMA	TrSistema	Sistema
→ MULTIPLE	String	Indica si el tipo de documento es múltiple o no. → “ S ” – Tipo de documento es múltiple → “ N ” – Tipo de documento no es múltiple
→ TEXTOAUXILIAR	String	Texto auxiliar de libre uso para las aplicaciones que utilizan TREW@
→ INFORMAR	String	Indica si se informa al bus del tipo de documento → “ S ” – Sí → “ N ” – No
→ VERSIONABLE	String	Indica si el tipo de documento es versionable o no. → “ S ” – Tipo de documento es versionable → “ N ” – Tipo de documento no es versionable
→ REUTILIZABLE	String	Indica si el tipo de documento es reutilizable o no. → “ S ” – Tipo de documento es reutilizable → “ N ” – Tipo de documento no es reutilizable
→ MODOGEN	String	Modo de generación del documento → “ R ” – Report Server Oracle → “ P ” – Generación PDF → “ O ” – OpenOffice Bean
→ FUSIONARVAR	String	Indica si se fusionan las variables al generar → “ S ” – Sí → “ N ” – No
→ FIRMADIGI	String	Indica si se puede firmar digitalmente. → “ S ” – Sí → “ N ” – No
→ TIPOFIRMA	String	Tipo de firma para el documento. → “ C ” – Cascada → “ P ” – Paralelo
→ REGISTRABLE	String	Indica si el tipo de documento es registrable o no. → “ S ” – Tipo de documento es registrable → “ N ” – Tipo de documento no es registrable
→ NOTIFICABLE	String	Indica si el tipo de documento es notificable o no. → “ S ” – Tipo de documento es notificable → “ N ” – Tipo de documento no es notificable
→ ARCHIVABLE	String	Indica si el tipo de documento es archivable o no. → “ S ” – Tipo de documento es archivable → “ N ” – Tipo de documento no es archivable
→ OBSOLETO	String	Indica si el tipo de documento está obsoleto.

→	CODWANDA	String	→ "S" – Sí
→	PLANTILLA	TrPlantilla	→ "N" – No
→	CLASEDOC	String	Valor en w@ndA del tipo de documento.
→	TIPO_CERTIF	TrTipoCertificado	Plantilla utilizada para la generación de documentos Report y Java PDF.
→	VALIDA_CONSULTA	String	Indica de qué clase es el documento.
→	CONSULTA_PLANTILLA_MULT	TrConsPlantMult	Sus valores pueden ser:
→	ADMINISTRATIVO	String	→ "G" – General
→	EDITABLE	String	→ "S" – SCSP con necesidad de consentimiento
→	FIRMA_DINAMICA	String	→ "L" – SCSP autorizado por Ley
→	INFORMACION_PUBLICICA	String	Tipo de certificado para los documentos SCSP.
→	TIPODOCENI	TrTipoDocumentoEni	Indica si la consulta de plantilla múltiple es válida o no.
→	TIPO_PAGO	TrTipoPago	→ "S" – Si
			→ "N" – No
			Consulta que se aplicará para calcular la plantilla múltiple del tipo documento al generar lo.
			Indica si es administrativo o auxiliar. Sus valores pueden ser:
			→ "S" – Si
			→ "N" – No
			Indica si se puede editar. Sus valores pueden ser:
			→ "S" – Si
			→ "N" – No
			Indica si el tipo de documento podrá ser firmado por firmantes no asociados al mismo o no. Sus valores pueden ser:
			→ "S" – Si
			→ "N" – No
			Indica si el tipo de documento tendrá visibilidad para información pública. Sus valores pueden ser:
			→ "S" – Si
			→ "N" – No
			Tipo de documento correspondiente al Esquema Nacional de Interoperabilidad (ENI) al que pertenece el tipo de documento.
			Indica el tipo de pago del tipo de documento

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento
→ CAMPO_ETIQUETA	CADENA(10)	Abreviatura del tipo de documento
→ CAMPO_NOMBRE	CADENA(200)	Nombre del tipo de documento
→ CAMPO_DESCRIPCION	CADENA	Descripción del tipo de documento
→ CAMPO_ENTRADASALIDA	CADENA(1)	Indica cómo está definido el documento en TREW@ → “E” – Entrada → “S” – Salida → “ES” – Entrada / Salida
→ CAMPO_INCGEN	CADENA(1)	Indica cómo está definido el documento → “I” – Documento a Incorporar al expediente → “G” – Documento a Generar
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
→ CAMPO_MULTIPLE	CADENA(1)	Indica si el tipo de documento es múltiple o no. → “S” – Tipo de documento es múltiple → “N” – Tipo de documento no es múltiple
→ CAMPO_TEXTOAUXILIAR	CADENA(50)	Texto auxiliar de libre uso para las aplicaciones que utilizan TREW@
→ CAMPO_INFORMAR	CADENA(1)	Indica si se informa al bus del tipo de documento → “S” – Sí → “N” – No
→ CAMPO_VERSIONABLE	CADENA(1)	Indica si el tipo de documento es versionable o no. → “S” – Tipo de documento es versionable → “N” – Tipo de documento no es versionable
→ CAMPO_REUTILIZABLE	CADENA(1)	Indica si el tipo de documento es reutilizable o no. → “S” – Tipo de documento es reutilizable → “N” – Tipo de documento no es reutilizable
→ CAMPO_MODALOGEN	CADENA(1)	Modo de generación del documento → “R” – Report Server Oracle → “P” – Generación PDF → “O” – OpenOffice Bean
→ CAMPO_FUSIONAR	CADENA(1)	Indica si se fusionan las variables al generar → “S” – Sí → “N” – No
→ CAMPO_FIRMADIGI	CADENA(1)	Indica si se puede firmar digitalmente. → “S” – Sí → “N” – No
→ CAMPO_TIPOFIRMA	CADENA(1)	Tipo de firma para el documento. → “C” – Cascada → “P” – Paralelo
→ CAMPO_FORMATO	CADENA(128)	Tipo de formato de la plantilla (tipo mime).
→ CAMPO_NOMBREFICHERO	CADENA(64)	Nombre del fichero de la plantilla.
→ CAMPO_REGISTRABLE	CADENA(1)	Indica si el tipo de documento es registrable o no. → “S” – Tipo de documento es registrable → “N” – Tipo de documento no es registrable
→ CAMPO_NOTIFICABLE	CADENA(1)	Indica si el tipo de documento es notificable o no. → “S” – Tipo de documento es notificable

→	CAMPO_ARCHIVABLE	CADENA(1)	→ "N" – Tipo de documento no es notificable Indica si el tipo de documento es archivable o no. → "S" – Tipo de documento es archivable
→	CAMPO_OBSOLETO	CADENA(1)	→ "N" – Tipo de documento no es archivable Indica si el tipo de documento está obsoleto. → "S" – Sí → "N" – No
→	CAMPO_CODWANDA	CADENA(30)	Valor en w@ndA del tipo de documento.
→	CAMPO_REFPLANTILLA	NUMÉRICO	Identificador de la plantilla para la generación de documentos Report o JavaPDF
→	CAMPO_NOMBREPLANTILLA	CADENA(15)	Nombre de la plantilla
→	CAMPO_DESCPLANTILLA	CADENA(250)	Descripción de la plantilla
→	CAMPO_NOMBINFOME	CADENA(15)	Nombre del informe de la plantilla
→	CAMPO_CLASEDOC	CADENA(20)	Indica de qué clase es el documento. Por ejemplo SCSP o GENERAL. Sus valores pueden ser: → "G" – General → "S" – SCSP con necesidad de consentimiento → "L" – SCSP autorizado por Ley
→	CAMPO_REFCONSULTA	NUMÉRICO	Identificador de la consulta para la generación de la plantilla múltiple del tipo de documento
→	CAMPO_CONSVALIDA	CADENA(1)	Indica si la consulta de la plantilla múltiple del tipo de documento es válida. → "S" – Sí → "N" – No
→	CAMPO_ADMINISTRATIVO	CADENA(1)	Indica si es administrativo o auxiliar. Sus valores pueden ser: → "S" – Sí → "N" – No
→	CAMPO_EDITABLE	CADENA(1)	Indica si se puede editar. Sus valores pueden ser: → "S" – Sí → "N" – No
→	CAMPO_INFORMACIONPUBLICA	CADENA(1)	Indica si el tipo de documento tendrá visibilidad para información pública. Sus valores pueden ser: → "S" – Sí → "N" – No
→	CAMPO_REFTIPDOCENI	NUMÉRICO	Identificador de tipo de documento ENI

- generarDocumentoMultiple
- importarDocumento
- incorporarDocExternoMultiplesExpedientes
- incorporarDocumento
- incorporarDocumentoExterno
- incorporarDocumentoMultiplesExpedientes
- incorporarDocumentoNoDefinido
- incorporarDocumentoNoDefinidoExterno
- obtenerDocExp
- obtenerDocumentosExpediente
- obtenerDocumentosExpedienteSimple
- obtenerDocumentosPermitidos
- obtenerFirmantesTipoDocumento
- obtenerParrafoTipoDocumento
- obtenerRelacionesDefinidas
- obtenerTipoDocIncorporar
- obtenerTipoDocumentoSimple
- obtenerTiposDocumento

Métodos para aplicación de los filtros y dónde se usa

2.4.1.2.92 TrTipoExpediente

trewa.bd.trapi.trapiui.tpo.TrTipoExpediente

Clase que representa la información referente a tipos de expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoría
→ REFTIPOEXP	TpoPK	Identificador del tipo de expediente
→ ABREVIATURA	String	Abreviatura del tipo de expediente
→ DESCRIPCION	String	Descripción del tipo de expediente
→ VIGENTE	String	Indica si el tipo de expediente está vigente en TREW@ → "S" – El tipo de expediente Sí está vigente → "N" – El tipo de expediente No está vigente
→ STMA	TrSistema	Sistema al que está asociado el tipo de expediente.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
--------------	-------------	---

→	CAMPO_REFTIPOEXP	NUMÉRICO	Identificador del tipo de expediente
→	CAMPO_ABREVIATURA	CADENA(10)	Abreviatura del tipo de expediente
→	CAMPO_DESCRIPCION	CADENA(50)	Descripción del tipo de expediente
→	CAMPO_VIGENTE	CADENA(1)	Indica si el tipo de expediente está vigente en TREW@ → “ S ” – El tipo de expediente Sí está vigente → “ N ” – El tipo de expediente No está vigente
→	CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema al que está asociado el tipo de expediente.
→	CAMPO_CREADO	CADENA	Usuario que crea el registro
→	CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→	CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→	CAMPO_FECHAMODIFICA	DATE	Fecha de modificación del registro

Métodos para aplicación de los filtros

→ obtenerTiposExpediente

2.4.1.2.93 TrTipoidentificador

trewa.bd.trapi.trapiui.tpo.TrTipoidentificador

Clase que representa la información de los distintos tipos de identificadores (DNI, NIF...).

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	CODTIPOIDENT	String	Código del tipo de identificador.
→	DESCRIPCION	String	Descripción del tipo de identificador.
→	TIPOIDENT	String	Indica si se trata de una persona física o jurídica. → “ F ” – Física → “ J ” – Jurídica
→	CODWANDA	String	Valor en w@ndA del tipo de identificador
→	OBSOLETO	String	Indica si el tipo de identificador está obsoleto. → “ S ” – Física → “ N ” – Jurídica

Campos definidos para filtrados y ordenación

	<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_CODTIPOIDENT	CADENA(10)	Código del tipo de identificador.
→	CAMPO_DESCRIPCION	CADENA(100)	Descripción del tipo de identificador.
→	CAMPO_TIPOIDENT	CADENA(1)	Indica si se trata de una persona física o jurídica. → “ F ” – Física → “ J ” – Jurídica
→	CAMPO_CODWANDA	CADENA(30)	Valor en w@ndA del tipo de identificador
→	CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de identificador está obsoleto. → “ S ” – Sí → “ N ” – No

Métodos para aplicación de los filtros

→ obtenerTipoidentificador

2.4.1.2.94 TrTipoIndicacion

trewa.bd.trapi.trapiui.tpo.TrTipoIndicacion

Clase que representa los tipos de datos variables sobre la ficha descriptiva del procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOIND	TpoPK	Identificador del tipo de indicación de la ficha.
→ ABREVIATURA	String	Abreviatura del tipo de indicación de la ficha descriptiva del procedimiento.
→ DESCRIPCION	String	Descripción del tipo de indicación de la ficha descriptiva del procedimiento.
→ CODWANDA	String	Valor en w@ndA del tipo de indicación de la ficha descriptiva del procedimiento.
→ OBSOLETO	String	Indica si el tipo de indicación está obsoleto. → " S " – Sí → " N " – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOIND	NUMÉRICO	Identificador del tipo de indicación de la ficha.
→ CAMPO_ABREVIATURA	CADENA(10)	Abreviatura del tipo de indicación de la ficha descriptiva del procedimiento.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción del tipo de indicación de la ficha descriptiva del procedimiento.
→ CAMPO_CODWANDA	CADENA(30)	Valor en w@ndA del tipo de indicación de la ficha descriptiva del procedimiento.
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de indicación está obsoleto. → " S " – Sí → " N " – No

Métodos para aplicación de los filtros

→ obtenerTipoIndicacion

2.4.1.2.95 TrTipoOrganismo

trewa.bd.trapi.trapiui.tpo.TrTipoOrganismo

Clase que representa la información de los distintos tipos de organismos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOORG	TpoPK	Identificador del tipo de organismo.
→ DESCRIPCION	String	Descripción del tipo de organismo.
→ ABREVIATURA	String	Abreviatura del tipo de organismo.
→ CODWANDA	String	Valor en w@ndA del tipo de organismo
→ OBSOLETO	String	Indica si el tipo de organismo está obsoleto. → “ S ” – Sí → “ N ” – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOORG	NUMÉRICO	Identificador del tipo de organismo.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción del tipo de organismo.
→ CAMPO_ABREVIATURA	CADENA(10)	Abreviatura del tipo de organismo.
→ CAMPO_CODWANDA	CADENA(1)	Valor en w@ndA del tipo de organismo
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de organismo está obsoleto. → “ S ” – Sí → “ N ” – No

Métodos para aplicación de los filtros

→ obtenerTipoOrganismo

2.4.1.2.96 TrTipoOrganizacion

trewa.bd.trapi.trapiui.tpo.TrTipoOrganizacion

Clase que representa la información de los distintos tipos de organización.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODTIPOORGZ	String	Identificador del tipo de organización.
→ DESCRIPCION	String	Descripción del tipo de organización.
→ VIGENTE	String	Indica si el tipo de organización está vigente. → “ S ” – Sí → “ N ” – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODTIPOORGZ	CADENA(1)	Identificador del tipo de organización.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción del tipo de organización.
→ CAMPO_VIGENTE	CADENA(1)	Indica si el tipo de organización está vigente. → “ S ” – Sí → “ N ” – No

Métodos para aplicación de los filtros

→ obtenerTipoOrganizacion

2.4.1.2.97 TrTipoPago

trewa.bd.trapi.trapiui.tpo.TrTipoPago

Clase que representa la información referente a un tipo de pago.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOPAGO	TpoPK	Identificador del tipo de pago.
→ MODELO	String	Modelo del tipo de pago.
→ DESCRIPCION	String	Descripción del tipo de pago.
→ CONCEPTO	String	Concepto del tipo de pago.

- | | | | |
|---|------------------------|---------|---|
| → | COD_TERRITORIA | Integer | Código territorial del tipo de pago. |
| | L | | |
| → | IMPORTE_DEFECT | Float | Importe por defecto del tipo de pago. |
| | O | | |
| → | IVA | Float | Porcentaje de IVA del tipo de pago. |
| → | PLAZO_PAGO | Integer | Plazo para realizar el tipo de pago. |
| → | AUTOLIQUIDACION | String | Indica si el tipo de pago es autoliquidable o no. |
| | N | | |
| | | | → “S” – El tipo de pago Sí es autoliquidable |
| | | | → “N” – El tipo de pago No es autoliquidable |
| → | OBLIGATORIO | String | Indica si el tipo de pago es obligatorio o no. |
| | | | → “S” – El tipo de pago Sí es obligatorio |
| | | | → “N” – El tipo de pago No es obligatorio |

Campos definidos para filtrados y ordenación

	<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→	CAMPO_REFTIPOPAGO	NUMÉRICO	Identificador del tipo de pago.
→	CAMPO_MODELO	CADENA(32)	Modelo del tipo de pago.
→	CAMPO_DESCRIPCION	CADENA(250)	Descripción del tipo de pago.
→	CAMPO_CONCEPTO	CADENA(128)	Concepto del tipo de pago.
→	CAMPO_CODTERRITORIAL	NUMÉRICO	Código territorial del tipo de pago.
→	CAMPO_IMPORTEDEFECTO	NUMÉRICO	Importe por defecto del tipo de pago.
→	CAMPO_IVA	NUMÉRICO	Porcentaje de IVA del tipo de pago.
→	CAMPO_PLAZOPAGO	NUMÉRICO	Plazo para realizar el tipo de pago.
→	CAMPO_AUTOLIQUIDACION	CADENA(1)	Indica si el tipo de pago es autoliquidable o no. → “S” – El tipo de pago Sí es autoliquidable → “N” – El tipo de pago No es autoliquidable
→	CAMPO_OBLIGATORIO	CADENA(1)	Indica si el tipo de pago es obligatorio o no. → “S” – El tipo de pago Sí es obligatorio → “N” – El tipo de pago No es obligatorio

Métodos para aplicación de los filtros

- obtenerPagosTasa
- obtenerTipoDocumento
- obtenerTipoDocumentoSimple
- obtenerTiposDocumento
- obtenerTiposPago
- obtenerTiposPagoProcedimiento

2.4.1.2.98 TrTipoPagoProcedimiento

trewa.bd.trapi.trapiui.tpo.TrTipoPagoProcedimiento

Clase que representa la información referente a un tipo de pago asociado al procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDEFPROC	TpoPK	Identificador del procedimiento.
→ TIOPAGO	TrTipoPago	Tipo de pago.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIOPAGO	NUMÉRICO	Identificador del tipo de pago.
→ CAMPO_MODELO	CADENA(32)	Modelo del tipo de pago.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del tipo de pago.
→ CAMPO_CONCEPTO	CADENA(128)	Concepto del tipo de pago.
→ CAMPO_CODTERRITORIAL	NUMÉRICO	Código territorial del tipo de pago.
→ CAMPO_IMPORTEDEFECTO	NUMÉRICO	Importe por defecto del tipo de pago.
→ CAMPO_IVA	NUMÉRICO	Porcentaje de IVA del tipo de pago.
→ CAMPO_PLAZOPAGO	NUMÉRICO	Plazo para realizar el tipo de pago.
→ CAMPO_AUTOLIQUIDACION	CADENA(1)	Indica si el tipo de pago es autoliquidable o no. → "S" – El tipo de pago Sí es autoliquidable → "N" – El tipo de pago No es autoliquidable
→ CAMPO_OBLIGATORIO	CADENA(1)	Indica si el tipo de pago es obligatorio o no. → "S" – El tipo de pago Sí es obligatorio → "N" – El tipo de pago No es obligatorio
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento

Métodos para aplicación de los filtros

- obtenerPagosTasa
- obtenerTiposPagoProcedimiento

2.4.1.2.99 TrTipoParrafo

trewa.bd.trapi.trapiui.tpo.TrTipoParrafo

Clase que representa la información referente a un tipo de párrafo.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOPARR	TpoPK	Identificador del tipo de párrafo
→ ABREVIATURA	String	Abreviatura del tipo de párrafo
→ DESCRIPCION	String	Descripción del tipo de párrafo
→ ETIQUETA	String	Etiqueta del tipo de párrafo
→ PARRAFO	String	Texto del tipo de párrafo
→ ALINEACION	String	Tipo de alineación del tipo de párrafo → “ I ” – A la izquierda → “ C ” – Centrada → “ D ” – A la derecha → “ J ” – Justificada
→ ESTILO	String	Estilo del tipo de párrafo
→ ESTILOETIQ	String	Estilo de la etiqueta del tipo de párrafo
→ UBICACION	String	Ubicación del tipo de párrafo → “ A ” – Cabecera → “ C ” – Cuerpo → “ P ” – Pie → “ O ” – Otro
→ EDITABLE	String	Indica si el tipo de párrafo es editable o no → “ S ” – El tipo de párrafo Sí es editable → “ N ” – El tipo de párrafo No es editable
→ STMA	TrSistema	Sistema al que pertenece el tipo de párrafo
→ NOMBREFICHERO	String	Nombre del fichero de la imagen
→ FORMATO	String	Formato o tipo mime del fichero
→ IMAGEN	byte[]	Imagen del tipo de párrafo
→ TIPOPARRAFOEDITOR	TrTipoParrafoEditor	Tipo de párrafo para el editor de párrafos

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOPARR	NUMÉRICO	Identificador del tipo de párrafo

→	CAMPO_ABBREVIATURA	CADENA(15)	Abreviatura del tipo de párrafo
→	CAMPO_DESCRIPCION	CADENA(250)	Descripción del tipo de párrafo
→	CAMPO_ETIQUETA	CADENA(15)	Etiqueta del tipo de párrafo
→	CAMPO_PARRAFO	CADENA(4000)	Texto del tipo de párrafo
→	CAMPO_ALINEACION	CADENA(1)	Tipo de alineación del tipo de párrafo → "I" – A la izquierda → "C" – Centrada → "D" – A la derecha → "J" – Justificada
→	CAMPO_ESTILO	CADENA(8)	Estilo del tipo de párrafo
→	CAMPO_ESTILOETIQ	CADENA(8)	Estilo de la etiqueta del tipo de párrafo
→	CAMPO_UBICACION	CADENA(1)	Ubicación del tipo de párrafo → "A" – Cabecera → "C" – Cuerpo → "P" – Pie → "O" – Otro
→	CAMPO_EDITABLE	CADENA(1)	Indica si el tipo de párrafo es editable o no → "S" – El tipo de párrafo Sí es editable → "N" – El tipo de párrafo No es editable
→	CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema al que pertenece el tipo de párrafo
→	CAMPO_NOMBREFICHERO	CADENA(64)	Nombre del fichero de la imagen
→	CAMPO_FORMATO	CADENA(128)	Formato o tipo mime del fichero

Métodos para aplicación de los filtros

- obtenerParrafosDocumento
- obtenerParrrafoTipoDocumento
- obtenerTiposParrrafo

2.4.1.2.100 TrTipoRelacion

trewa.bd.trapi.trapiui.tpo.TrTipoRelacion

Clase que representa la información referente a un tipo de relación.

Atributos accesibles mediante métodos get/set

	<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→	REFTIPORELA	TpoPK	Identificador del tipo de relación
→	NOMBRE	String	Nombre del tipo de relación
→	DESCRIPCION	String	Descripción del tipo de relación
→	STMA	TrSistema	Sistema

Métodos para aplicación de los filtros

- obtenerRelacionesDefinidas

2.4.1.2.101 TrTipoVia

trewa.bd.trapi.trapiui.tpo.TrTipoVia

Clase que representa la información de los tipos de vía.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODTIPOVIA	String	Identificador del tipo de vía.
→ DESCRIPCION	String	Descripción del tipo de vía.
→ CODWANDA	String	Valor en w@ndA del tipo de vía
→ OBSOLETO	String	Indica si el tipo de vía está obsoleto. → “ S ” – Sí → “ N ” – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODTIPOVIA	CADENA(10)	Identificador del tipo de vía.
→ CAMPO_DESCRIPCION	CADENA(32)	Descripción del tipo de vía.
→ CAMPO_CODWANDA	CADENA(15)	Valor en w@ndA del tipo de vía
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de vía está obsoleto. → “ S ” – Sí → “ N ” – No

Métodos para aplicación de los filtros

- obtenerEmpleado
- obtenerOrganismo
- obtenerOrganismosHijos
- obtenerTiposVia

2.4.1.2.102 TrTransicion

trewa.bd.trapi.trapiui.tpo.TrTransicion

Clase que representa la información referente a un transición.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTRANSICION	TpoPK	Identificador de la transición
→ ETIQUETA	String	Etiqueta de la transición
→ DESCRIPCION	String	Descripción de la transición
→ TIPOACTO	TrTipoActo	Tipo de acto que representa la transición
→ FASEFIN	TrFase	Fase final
→ ETIQUETALARGA	String	Etiqueta larga para la transición
→ TIPO	String	Tipo de la transición → “ N ” – Transición normal → “ D ” – Transición de división → “ U ” – Transición de unión → “ ES ” – Evento que provoca salida → “ EN ” – Evento que no provoca la salida
→ DESCFECHA	String	Descripción de lo que significa la fecha de realización de la transición
→ PLAZO	TrPlazo	Plazo de realización
→ REFTRANPROV	TpoPK	Identificador de la transición provocada por finalizar el plazo
→ NUMMAX	Integer	Número máximo de posibles repeticiones para la transición en el procedimiento
→ ORDEN	Integer	Orden entre las posibles transiciones que se pueden dar desde una determinada fase
→ INFORMARBUS	String	Indica si se informa al bus del cambio de fase → “ S ” – Sí → “ N ” – No
→ REFFASEINI	TpoPK	Identificador de la fase inicial
→ REFTRANSICIONPADRE	TpoPK	Identificador de la transición padre

Campos definidos para filtrados y ordenación



<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición
→ CAMPO_ETIQUETA	CADENA(11)	Etiqueta de la transición
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción de la transición
→ CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto que representa la transición
→ CAMPO_ABREVTIPOACTO	CADENA(10)	Abreviatura del tipo de acto
→ CAMPO_DESCTIPOACTO	CADENA(200)	Descripción del tipo de acto
→ CAMPO_ETIQUETALARGA	CADENA(25)	Etiqueta larga de la transición
→ CAMPO_TIPO	CADENA(2)	Tipo de la transición → "N" – Transición normal → "D" – Transición de división → "U" – Transición de unión → "ES" – Evento que provoca salida → "EN" – Evento que no provoca la salida
→ CAMPO_DESCFECHA	CADENA(50)	Descripción de lo que significa la fecha de realización de la transición
→ CAMPO_REFFASEFIN	NUMÉRICO	Identificador de la fase fin de la transición
→ CAMPO_REFFASEINI	NUMÉRICO	Identificador de la fase inicial de la transición
→ CAMPO_UNIDAD	CADENA(1)	Unidad de tiempo → "D" – Días → "M" – Meses → "A" – Años
→ CAMPO_NUMUNIDADES	NUMÉRICO	Número de unidades de tiempo
→ CAMPO_DESCFECHALIMIT	CADENA(100)	Descripción de la fecha límite
E		
→ CAMPO_REFTRANPROV	NUMÉRICO	Identificador de la transición provocada por la fecha límite
→ CAMPO_NUMMAX	NUMÉRICO	Número máximo de posibles repeticiones para la transición en el procedimiento
→ CAMPO_ORDEN	NUMÉRICO	Orden de la transición
→ CAMPO_INFOMARBUS	CADENA(1)	Indica si se informa al bus del cambio de fase → "S" – Sí → "N" – No
→ CAMPO_REFTRANSICIONPADRE	NUMÉRICO	Identificador de la transición padre

Métodos para aplicación de los filtros

- obtenerAvisosCaducidadExpediente
- obtenerAvisosCaducidadTransicion
- obtenerDatosTransicion
- obtenerFasesExpediente
- obtenerRelacionesDefinidas
- recorrerFasesPosTipEv
- recorrerEvolExpte
- recorrerTramitarExp

2.4.1.2.103 TrTransicionDefProcedimiento

trewa.bd.trapi.trapiui.tpo.TrTransicionDefProcedimiento

Clase que representa la información referente a una transición de un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ TRANSICION	TrTransicion	Transición
→ REFFASEPADRE	TpoPK	Para fases pertenecientes a módulos reutilizables, indica la fase que representa dicho procedimiento.
→ DEFPROC	TrDefProcedimiento	Si la transición pertenece a un módulo reutilizable, contiene la definición del procedimiento que representa dicho módulo
→ NUMMAX	Integer	Número máximo de veces por la que se puede pasar por una transición en un procedimiento
→ REFTRANPROV	TpoPK	Identificador de la transición provocada por la fecha límite

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición
→ CAMPO_ETIQUETA	CADENA(11)	Etiqueta de la transición
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción de la transición
→ CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto que representa la transición
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase en la que se quedaría el expediente con la transición
→ CAMPO_FASE	CADENA(50)	Nombre de la fase o situación en la que quedaría el expediente con la transición
→ CAMPO_METAFASE	CADENA(50)	Nombre la metafase a la que corresponde la fase en la que quedaría el expediente con la transición
→ CAMPO_UNIDAD	CADENA(1)	Unidad de tiempo → "D" – Días → "M" – Meses → "A" – Años

→	CAMPO_NUMUNIDADES	NUMÉRICO	Número de unidades de tiempo
→	CAMPO_DESCFECHALIM	CADENA(100)	Descripción de la fecha límite
→	CAMPO_REFTRANPROV	NUMÉRICO	Identificador de la transición provocada por la fecha límite
→	CAMPO_NUMMAX	NUMÉRICO	Número máximo de posibles repeticiones para la transición en el procedimiento
→	CAMPO_DESCFASE	CADENA(250)	Descripción de la fase final de la transición
→	CAMPO_DESCFECHA	CADENA(50)	Descripción de la fecha de la transición
→	CAMPO_DESCMETAFASE	CADENA(250)	Descripción de la metafase a la que corresponde la fase en la que quedaría el expediente con la transición
→	CAMPO_ETIQUETALARGA	CADENA(25)	Etiqueta larga de la transición
→	CAMPO_INFORMARFASE	CADENA(1)	Indica si se informa la fase final de la transición al bus. Sus valores pueden ser: → "S" – Sí → "N" – No
→	CAMPO_INFORMARMETAFASE	CADENA(1)	Indica si se informa la metafase de la fase final de la transición al bus. Sus valores pueden ser: → "S" – Sí → "N" – No
→	CAMPO_INFORMARTRANS	CADENA(1)	Indica si se informa la transición al bus. Sus valores pueden ser: → "S" – Sí → "N" – No
→	CAMPO_ORDEN	NUMÉRICO	Orden de la transición
→	CAMPO_ORDENFASE	NUMÉRICO	Orden de la fase final de la transición
→	CAMPO_ORDENMETAFASE	NUMÉRICO	Orden de la metafase a la que corresponde la fase en la que quedaría el expediente con la transición.
→	CAMPO_REFDEFPROCREUT	NUMÉRICO	Identificador del procedimiento reutilizable que representa la fase final de la transición
→	CAMPO_REFFASEINI	NUMÉRICO	Identificador de la fase de inicio de la transición
→	CAMPO_REFMETAFASE	NUMÉRICO	Identificador de la metafase a la que corresponde la fase en la que quedaría el expediente con la transición
→	CAMPO_REFSTMAFASE	NUMÉRICO	Identificador del sistema de la fase final de la transición
→	CAMPO_REFSTMAMETAFASE	NUMÉRICO	Identificador del sistema de la metafase a la que corresponde la fase en la que quedaría el expediente con la transición
→	CAMPO_REFTRANSICIONPADRE	NUMÉRICO	Identificador de la transición padre de la transición
→	CAMPO_TEXTOAUXILIARFASE	CADENA(500)	Texto auxiliar de la fase final de la transición
→	CAMPO_TIPO	CADENA(2)	Tipo de la transición
	Métodos para aplicación de los filtros		→ obtenerTransicionesDefProcedimiento

2.4.1.2.104 TrTransicionExplorador

trewa.bd.trapi.trapiui.tpo.TrTransicionExplorador

Clase que representa la información referente a un transición pendiente en una fase de un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTRANSICION	TpoPK	Identificador de la transición
→ DESCRIPCION	String	Descripción de la transición
→ FASEFIN	TrFase	Fase final
→ TIPO	String	Tipo de la transición → “ N ” – Transición normal → “ D ” – Transición de división → “ U ” – Transición de unión → “ ES ” – Evento que provoca salida → “ EN ” – Evento que no provoca la salida
→ REFFASEINI	TpoPK	Identificador de la fase inicial
→ EXPFASE	TpoPK	Expediente en fase de la transición. Utilizado para saber si una transición pertenece a reutilizables o no.

Métodos para aplicación de los filtros

→ explorarExpedientes

2.4.1.2.105 TrUsuario

trewa.bd.trapi.trapiui.tpo.TrUsuario

Clase que representa la información del usuario.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoría
→ CODUSUARIO	String	Identificador del usuario
→ TIPOIDENT	String	Tipo de identificador. → “ D ” – DNI / NIF → “ P ” – Pasaporte → “ N ” – NIE → “ C ” – CIF → “ O ” – Otros
→ IDENTIFICADOR	String	Código de identificación
→ NOMBRE	String	Nombre del usuario
→ APELLIDO1	String	Primer apellido del usuario.
→ APELLIDO2	String	Segundo apellido del usuario.
→ SEXO	String	Sexo del usuario. → “ F ” – Femenino → “ M ” – Masculino
→ EMAIL	String	Dirección de correo electrónico del usuario.
→ CLAVE	String	Clave del usuario (típicamente encriptada).
→ FECHAALTA	Timestamp	Fecha de alta en el sistema.
→ FECHABAJA	Timestamp	Fecha de baja en el sistema.

- **ANAGRAMAFISCA** String Anagrama fiscal del usuario
- L**
- **FIRMA** TrFirma Datos de la firma del usuario
- **FECHAMODCLAVE** Timestamp Fecha de la última modificación de clave de usuario

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODUSUARIO	CADENA(10)	Identificador del usuario
→ CAMPO_TIPOIDENT	CADENA(1)	Tipo de identificador. → “ D ” – DNI / NIF → “ P ” – Pasaporte → “ N ” – NIE → “ C ” – CIF → “ O ” – Otros
→ CAMPO_IDENTIFICADOR	CADENA(15)	Código de identificación
→ CAMPO_NOMBRE	CADENA(32)	Nombre del usuario
→ CAMPO_APELLIDO1	CADENA(32)	Primer apellido del usuario.
→ CAMPO_APELLIDO2	CADENA(32)	Segundo apellido del usuario.
→ CAMPO_SEXO	CADENA(1)	Sexo del usuario. → “ F ” – Femenino → “ M ” – Masculino
→ CAMPO_EMAIL	CADENA(64)	Dirección de correo electrónico del usuario.
→ CAMPO_CLAVE	CADENA(32)	Clave del usuario (típicamente encriptada).
→ CAMPO_FECHAALTA	DATE	Fecha de alta en el sistema.
→ CAMPO_FECHABAJA	DATE	Fecha de baja en el sistema.
→ CAMPO_ANAGRAMAFISCAL	VARCHAR2(50)	Anagrama fiscal del usuario
→ CAMPO_REFFIRMA	NUMÉRICO	Identificador de la firma del usuario
→ CAMPO_FORMATO	VARCHAR2(128)	Formato de la imagen de la firma
→ CAMPO_NOMBREFICHERO	VARCHAR2(64)	Nombre del fichero de la imagen de la firma
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICA	DATE	Fecha de modificación del registro
→ CAMPO_FECHACLAVE	DATE	Fecha de última modificación de clave de usuario

Métodos para aplicación de los filtros

- obtenerEmpleado
- obtenerPerfilesUsuario
- obtenerUsuario
- obtenerUsuarioAsignado

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.1.2.106 TrUsuarioAsignado

trewa.bd.trapi.trapiui.tpo.TrUsuarioAsignado

Clase que representa la información de los usuarios asignados al expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFUSUASIGNADO	TpoPK	Identificador del usuario asignado
→ USUARIO	TrUsuario	Usuario
→ TIPO	String	Tipo de usuario asignado. → “ P ” – Principal → “ S ” – Secundario
→ FECHAALTA	Timestamp	Fecha de asignación al expediente.
→ FECHABAJA	Timestamp	Fecha de baja.
→ RAZONASIGNACION	String	Razón de la asignación.
→ REFEXPEDIENTE	TpoPK	Identificador del expediente
→ EXPEDIENTE	TrExpediente	Expediente.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFUSUASIGNADO	NUMERICO	Identificador del usuario asignado
→ CAMPO_USUARIO	CADENA(10)	Usuario
→ CAMPO_TIPO	CADENA(1)	Tipo de usuario asignado. → “ P ” – Principal → “ S ” – Secundario
→ CAMPO_FECHAALTA	DATE	Fecha de asignación al expediente.
→ CAMPO_FECHABAJA	DATE	Fecha de baja.
→ CAMPO_RAZONASIGNACION	CADENA(250)	Razón de la asignación.

→	CAMPO_REFEXPEDIENTE	NUMÉRICO	Identificador del expediente
→	CAMPO_TIPOIDENT	CADENA(1)	Tipo de identificador del usuario → “ D ” – DNI / NIF → “ P ” – Pasaporte → “ N ” – NIE → “ C ” – CIF → “ O ” – Otros
→	CAMPO_IDENTIFICADOR	CADENA(15)	Código de identificación
→	CAMPO_NOMBRE	CADENA(32)	Nombre del usuario
→	CAMPO_APELLIDO1	CADENA(32)	Primer apellido del usuario.
→	CAMPO_APELLIDO2	CADENA(32)	Segundo apellido del usuario.
→	CAMPO_SEXO	CADENA(1)	Sexo del usuario. → “ F ” – Femenino → “ M ” – Masculino
→	CAMPO_EMAIL	CADENA(64)	Dirección de correo electrónico del usuario.
→	CAMPO_CLAVE	CADENA(32)	Clave del usuario (típicamente encriptada).
→	CAMPO_FECHAALTAUSUARIO	DATE	Fecha de alta en el sistema.
→	CAMPO_FECHABAJAUSUARIO	DATE	Fecha de baja en el sistema.
→	CAMPO_ANAGRAMAFISCAL	VARCHAR2(50)	Anagrama fiscal del usuario
→	CAMPO_INFORMADO	CADENA(1)	Indica si se ha informado al bus del expediente Sus valores pueden ser: - “ S ” - Sí - “ N ” – No
→	CAMPO Borrado	CADENA(1)	Indica si el expediente está borrado Sus valores pueden ser: - “ S ” - Sí - “ N ” – No
→	CAMPO_NUMEXP	CADENA(30)	Número del expediente
→	CAMPO_TITULOEXP	CADENA(250)	Título del expediente
→	CAMPO_OBSERVACIONES	CADENA(250)	Observaciones del expediente
→	CAMPO_PROPIETARIOEXP	CADENA(10)	Usuario propietario del expediente
→	CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo al que pertenece el expediente
→	CAMPO_REFORGENVIA	NUMÉRICO	Identificador del organismo que envía el expediente
→	CAMPO_URLWANDA	CADENA(255)	Url del expediente en w@ndA
→	CAMPO_OTROSDATOS	CADENA(1)	Indica si el expediente tiene otros datos Sus valores pueden ser: - “ S ” - Sí - “ N ” – No
→	CAMPO_FECHAARCHIVO	DATE	Fecha de archivo del expediente
→	CAMPO_REFCOMPONENTE	NUMÉRICO	Identificador del componente donde está archivado el expediente
→	CAMPO_FECHACLAVE	DATE	Fecha de última modificación de clave de usuario

Métodos para aplicación de los filtros

→ obtenerUsuarioAsignado

2.4.1.2.107 TrUsuarioAsignadoExplorador

trewa.bd.trapi.trapiui.tpo.TrUsuarioAsignadoExplorador

Clase que representa la información de un usuario asignado a un expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ NOMBRE	TpoPK	Nombre del usuario asignado.
→ APELLIDO1	TpoPK	Primer apellido del usuario asignado.
→ APELLIDO2	String	Segundo apellido del usuario asignado.

Métodos para aplicación de los filtros

→ explorarExpedientes

2.4.1.2.108 TrUsuarioPerfilUsuario

trewa.bd.trapi.trapiui.tpo.TrUsuarioPerfilUsuario

Clase que representa un perfil de usuario asignado a un usuario.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ PERFILUSU	TrPerfilUsuario	Perfil de usuario.
→ USUARIO	String	Código del usuario.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPERFILUSU	NUMÉRICO	Identificador del perfil de usuario.
→ CAMPO_NOMBREPERFIL	CADENA(25)	Nombre del perfil de usuario.
→ CAMPO_DESCPERFIL	CADENA(250)	Descripción del perfil de usuario.
→ CAMPO_USUARIO	CADENA(10)	Código del usuario.
→ CAMPO_TIPOIDENTUSU	CADENA(1)	Tipo de identificador. Sus valores pueden ser: <ul style="list-style-type: none"> - "D" – DNI / NIF - "P" – Pasaporte - "N" – NIE - "C" – CIF - "O" – Otros
→ CAMPO_IDENTIFICADOR USU	CADENA(15)	Código de identificación del usuario.
→ CAMPO_NOMBREUSU	CADENA(32)	Nombre del usuario.

→ CAMPO_APELLIDO1USU	CADENA(32)	Primer apellido del usuario.
→ CAMPO_APELLIDO2USU	CADENA(32)	Segundo apellido del usuario.
→ CAMPO_SEXOUSU	CADENA(1)	Sexo del usuario. Sus valores pueden ser:
		- "F" – Femenino
		- "M" – Masculino
→ CAMPO_EMAILUSU	CADENA(64)	Dirección de correo electrónico del usuario.
→ CAMPO_CLAVEUSU	CADENA(32)	Clave del usuario (típicamente encriptada)
→ CAMPO_FECHAALTAUSU	DATE	Fecha de alta del usuario en el sistema.
→ CAMPO_FECHABAJAUSU	DATE	Fecha de baja del usuario en el sistema.
→ CAMPO_ANAGRAMAFISCALUSU	CADENA(50)	Anagrama fiscal del usuario.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema del perfil de usuario.
→ CAMPO_CODSTMA	CADENA(10)	Código del sistema del perfil de usuario.
→ CAMPO_DESCSTMA	CADENA(128)	Descripción del sistema del perfil de usuario.
→ CAMPO_CONEXIONBUS	CADENA(1)	Indica si el sistema tiene conexión al bus. Sus valores pueden ser
		- "S" – Sí
		- "N" – No
→ CAMPO_REFCOMPINF	NUMÉRICO	Identificador del componente a informar del sistema del perfil de usuario.
→ CAMPO_REFCOMPVI	NUMÉRICO	Identificador del componente del avisador del sistema del perfil de usuario
→ CAMPO_REFCOMPGUAR	NUMÉRICO	Identificador del componente que guarda los documentos del sistema del perfil de usuario.
→ CAMPO_REFCOMPARCH	NUMÉRICO	Identificador del componente que archiva del sistema del perfil de usuario.
→ CAMPO_REFCOMPREG	NUMÉRICO	Identificador del componente que registra los documentos del sistema del perfil de usuario.
→ CAMPO_REFCOMPFIRMA	NUMÉRICO	Identificador del componente que firma los documentos del sistema del perfil de usuario.
→ CAMPO_REFCOMPNOTI	NUMÉRICO	Identificador del componente que notifica los documentos del sistema del perfil de usuario.
→ CAMPO_FECHACLAVE	DATE	Fecha de la última modificación de clave de usuario

Métodos para aplicación de los filtros

→ obtenerPerfilesUsuario

2.4.1.2.108.1 TrPlantillaTipoDocumento

trewa.bd.trapi.tpo.TrPlantillaTipoDocumento

Clase que representa las plantillas open office usadas en la generación de documentos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ COD_PLANT_OFFIC	String	Código de la plantilla office.

E

→ REFTIPODOC	TpoPK	Identificador del tipo de documento.
→ NOMBRE	String	Nombre del fichero.
→ FORMATO	String	Formato del fichero.
→ PLANTILLAOFFICE	Byte[]	Contenido del fichero.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento
→ CAMPO_CODPLANOFFICE	CADENA(64)	Identificador de la plantilla office del tipo documento
→ CAMPO_NOMBRE	CADENA(64)	Nombre de la plantilla office
→ CAMPO_FORMATO	CADENA(128)	Formato de la plantilla office

Métodos para aplicación de los filtros

→ obtenerPlantillaTipoDocumento

2.4.1.2.109 TrValorParametro

trewa.bd.trapi.trapiui.tpo.TrValorParametro

Clase que representa la información referente a un valor de un parámetro.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ PARAMETRO	String	Nombre del parámetro
→ VALOR	String	Valor del parámetro

2.4.1.2.110 TrVariable

trewa.bd.trapi.trapiui.tpo.TrVariable

Clase que representa la información referente a una variable de un sistema.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFVARIABLE	TpoPK	Identificador de la variable
→ NOMBRE	String	Nombre de la variable
→ DESCRIPCION	String	Descripción de la variable
→ FUNCION	String	Nombre de la función
→ STMA	TrSistema	Sistema
→ TIPOACTO	TrTipoActo	Tipo de acto
→ PAQUETE	String	Nombre del paquete que contiene la función almacenada o método java.
→ IMPLEMENTACION	String	Tipo de implementación
N		→ "F" – Implementación PL-SQL → "J" – Implementación Java

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFVARIABLE	NUMÉRICO	Identificador de la variable
→ CAMPO_NOMBRE	CADENA(25)	Nombre de la variable
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la variable
→ CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto
→ CAMPO_FUNCION	CADENA(50)	Nombre de la función
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
→ CAMPO_PAQUETE	CADENA(100)	Nombre del paquete que contiene la función almacenada o método java.
→ CAMPO_IMPLEMENTACION	CADENA(1)	Tipo de implementación → "F" – Implementación PL-SQL → "J" – Implementación Java
→ CAMPO_ABREVTIPOACTO	CADENA(10)	Abreviatura del tipo de acto administrativo
→ CAMPO_DESCTIPOACTO	CADENA(200)	Descripción del tipo de acto

Métodos para aplicación de los filtros

→ obtenerVariablesSistema

2.4.1.2.111TrVariableDocExp

trewa.bd.trapi.trapiui.tpo.TrVariableDocExp

Clase que recoge las variables del documento. Sólo válido para plantillas de Open Office.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFVARIABLE	TpoPK	Identificador de la variable
→ NOMBRE	String	Nombre de la variable
→ DESCRIPCION	String	Descripción de la variable

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFVARIABLE	NUMÉRICO	Identificador de la variable
→ CAMPO_NOMBRE	CADENA(25)	Nombre de la variable
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la variable

Métodos para aplicación de los filtros

→ obtenerVariableDocumentoExp

2.4.1.2.112 TrVersionDefProcedimiento

trewa.bd.trapi.trapiui.tpo.TrVersionDefProcedimiento

Clase que representa una versión de procedimiento asociada a un tipo de expediente determinado.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ VERSION	String	Descripción de la versión
→ FECHAVIGOR	Timestamp	Fecha desde la cual está en vigor la versión del flujo o más concretamente la normativa a la que está asociada
→ DEFPROC	TrDefProcedimiento	Definición de procedimiento que tiene asociado la versión
→ TIPOEXP	TrTipoExpediente	Tipo de expediente al que está asociado la versión
→ STMA	TrSistema	Sistema

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_VERSION	CADENA(50)	Descripción de la versión

→	CAMPO_FECHAVIGOR	DATE	Fecha desde la cual está en vigor la versión del flujo o más concretamente la normativa a la que está asociada
→	CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición de procedimiento que tiene asociado la versión
→	CAMPO_ABREVIATURA	CADENA(10)	Abreviatura de la definición de procedimiento asociado
→	CAMPO_DESCDEFPROC	CADENA(50)	Descripción de la definición de procedimiento asociado
→	CAMPO_REFTIPOEXP	NUMÉRICO	Identificador del tipo de expediente al que está asociada la versión
→	CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema de la versión
→	CAMPO_DESCAMPDEFPROC	CADENA(250)	Descripción ampliada del procedimiento
→	CAMPO_CODWANDA	CADENA(15)	Código w@ndA
→	CAMPO_VIGENTEDEFPROC	CADENA(1)	Indica si el procedimiento está vigente → "S" – Sí → "N" – No
→	CAMPO_ABREVIATURATIPOEXP	CADENA(10)	Abreviatura del tipo de expediente
→	CAMPO_DESCTIPOEXP	CADENA(50)	Descripción del tipo de expediente
→	CAMPO_VIGENTETIPOEXP	CADENA(1)	Indica si el tipo de expediente está vigente → "S" – Sí → "N" – No
→	CAMPO_INFOMAR	CADENA(1)	Indica si se informa al bus del procedimiento y de los expedientes que se tramiten

Métodos para aplicación de los filtros

- obtenerTipoExpAndTipoEvVersionDefProc
- obtenerVersionesDefProcedimiento

2.4.1.2.113 TrIndiceElectronico

trewa.bd.trapi.trapiui.tpo.TrIndiceElectronico

Clase que representa la información referente a un índice electrónico.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ IDENTIFICADORENI	String	Identificador del expediente electrónico ENI. Garantiza la unicidad del expediente a nivel nacional
→ HUELLA	String	Valor de la huella
→ FUNCIONRESUMEN	String	Identificador del expediente al que pertenece el documento
→ ORDEN	int	Orden del documento
→ FECHA	Timestamp	Fecha de generación o incorporación

2.4.1.2.114 TrIdentificadorEni

trewa.bd.trapi.trapiui.tpo.TrIdentificadorEni

Clase que representa la información referente a un identificador ENI.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ IDENTIFICADORENI	String	Identificador del expediente o documento electrónico ENI. Garantiza la unicidad del expediente o documento a nivel nacional.
→ IDIOMA	String	Abreviatura del idioma del identificador.
→ ORGANO	String	Identificador normalizado de la administración generadora del documento o expediente que realiza la captura del mismo. Este dato será extraído del Directorio Común de unidades orgánicas y oficinas.
→ FECHA	String	Indica el año de fecha de captura.
→ IDESPECIFICO	String	Código alfanumérico de 30 caracteres que identifica de forma única cada documento o expediente dentro de la organización responsable.

2.4.1.3 Servlets de subida y descarga de documentos

Para la subida y descarga de documentos, junto con la TrAPIUI, se aportan dos servlets que interactúan con la misma.

Para que los servlets de subida y descarga de documentos funcionen, se debe crear con antelación un objeto de la clase TrAPIUI y ponerlo a nivel de la sesión en la que estamos trabajando (**apiUI**).

2.4.1.3.1 Servlet DescargaDocumento

El Servlet DescargaDocumento inicia la descarga de un documento haciendo uso de métodos del TrAPIUI.

Los parámetros del servlet se indican en la siguiente tabla:

Parámetro	Ámbito	Ob.	Descripción
docExp	Request	Sí	Identificador del documento del expediente a descargar
multiple	Request	No	Indica si el documento es múltiple ("S") o no ("N"). El valor por defecto es "N".
pieFirma	Request	No	Indica si al documento se le incluye el pie de firma ("S") o no ("N"). El valor por defecto es "N".
versionados	Request	No	Indica si se obtienen los documentos versionados en el mismo documento. Si se indica "S" obtiene sólo los versionados (estado V), si se indica "N" se obtienen los no versionados (estado distinto de V) y si se indica "T" se obtienen todos. El valor por defecto es "T".
apiUI	Session	Sí/No	Interfaz TrAPIUI. Opcional si se indican los siguientes atributos en sesión.

En caso de que no se encuentre el apiUI en sesión, se pueden indicar los siguientes parámetros para crear la instancia del TrAPIUI.

trPerfilConexion	Session	Sí/No	Indica el perfil con el que se creará el TrAPIUI. Obligatorio siempre que no se encuentre el apiUI en sesión.
trUsuarioConexion	Session	No	Indica el usuario que se utilizará para crear la conexión con Trew@. Obligatorio si el trPerfilConexion indicado no está configurado con usuario y clave.
trClaveConexion	Session	No	Indica la clave del usuario que se utilizará para crear la conexión con Trew@. Obligatorio si el trPerfilConexion indicado no está configurado con usuario y clave.
trSistemaEstablecer	Session	No	Indica el código del sistema a establecer en la creación del TrAPIUI. Si no se indica se establecerá el sistema por defecto.
trUsuarioEstablecer	Session	No	Indica el usuario que se establecerá en el TrAPIUI una vez creada. Si no se indica se queda establecido el usuario de la conexión.

Ejemplo de llamada al servlet

→ Llamada desde una URL
<http://servidorWeb/aplicacionJSP/trewa/DescargaDoc?docExp=123>

→ Llamada desde un javascript
location.href='trewa/DescargaDoc?docExp=123'

2.4.1.3.2 Servlet SubidaDocumento

El Servlet SubidaDocumento sube un documento a la base de datos haciendo uso de métodos del TrAPIUI.

Los parámetros del servlet se indican en la siguiente tabla:

Parámetro	Ámbito	Ob.	Descripción
doc	Request	Sí	Identificador del documento a subir
apiUI	Session	Sí/No	Interfaz TrAPIUI. Opcional si se indican los siguientes atributos en sesión.

En caso de que no se encuentre el apiUI en sesión, se pueden indicar los siguientes parámetros para crear la instancia del TrAPIUI.

trPerfilConexion	Session	Sí/No	Indica el perfil con el que se creará el TrAPIUI. Obligatorio siempre que no se encuentre el apiUI en sesión.
trUsuarioConexion	Session	No	Indica el usuario que se utilizará para crear la conexión con Trew@. Obligatorio si el trPerfilConexion indicado no está configurado con usuario y clave.
trClaveConexion	Session	No	Indica la clave del usuario que se utilizará para crear la conexión con Trew@. Obligatorio si el trPerfilConexion indicado no está configurado con usuario y clave.
trSistemaEstablecer	Session	No	Indica el código del sistema a establecer en la creación del TrAPIUI. Si no se indica se establecerá el sistema por defecto.
trUsuarioEstablecer	Session	No	Indica el usuario que se establecerá en el TrAPIUI una vez creada. Si no se indica se queda establecido el usuario de la conexión.

Ejemplo de llamada al servlet

→ Llamada desde el action del form en JSP
 <FORM ENCTYPE="multipart/form-data" ACTION="trewa/SubidaDoc?doc=1246" METHOD="POST">

2.4.1.4 Métodos de la TrAPIUI

En este apartado se describe el conjunto de métodos existente en la TrAPIUI, los cuáles se han clasificado acorde a la tipología y misión de cada uno, agrupando aquellos que están relacionados según la funcionalidad que ofrecen.

Todos los métodos son accesibles a través de la interfaz **TrAPIUI**

2.4.1.4.1 APIs sobre documentos

2.4.1.4.1.1 Incorporar físicamente un documento

```
void adjuntarFicheroDocumento(
    TpoPK idDocExp,
    InputStream fichero,
    String nombreFichero,
    String formato,
    long size) throws TrException
```

Permite añadir un fichero físico al documento "incorporado" al expediente que se indica como parámetro.

Si se ha definido un componente w@rdA en el sistema establecido en el TrAPIUI y el documento ya estaba recogido en el mismo componente w@rdA, se creará el anexo en w@rdA y se asociará al documento.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento incorporado al expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>InputStream fichero</i>	<i>Fichero físico que se va a adjuntar al documento.</i>
<i>String nombreFichero</i>	<i>Nombre del fichero.</i>
<i>String formato</i>	<i>Tipo "mime" del fichero.</i>
<i>long size</i>	<i>Tamaño del fichero</i>

```
void adjuntarFicheroDocumento(
    TpoPK idDocExp,
    InputStream fichero,
    String nombreFichero,
    String formato,
    String idWarda, String idWardaAnx, TpoPK idComp) throws TrException
```

Permite añadir un fichero físico al documento “incorporado” y al expediente que se indica como parámetro.

Si se ha definido un componente w@rdA en el sistema establecido en el TrAPIUI y el documento ya estaba recogido en el mismo componente w@rdA, se creará el anexo en w@rdA y se asociará al documento.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento incorporado al expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>InputStream fichero</i>	<i>Fichero físico que se va a adjuntar al documento.</i>
<i>String nombreFichero</i>	<i>Nombre del fichero.</i>
<i>String formato</i>	<i>Tipo “mime” del fichero.</i>
<i>String idWarda</i>	<i>Identificador en w@rdA</i>
<i>String idWardaAnx</i>	<i>Identificador del anexo en w@rdA</i>
<i>TpoPK idComp</i>	<i>Identificador del componente w@rdA</i>

2.4.1.4.1.2 Actualizar documentos múltiples

```
void actualizarDocumentosMultiples(
    TpoPK idDocExpte,
    String modo) throws TrException
```

Permite actualizar el contenido de los párrafos y las firmas del documento que se pasa como parámetro al resto de documentos del mismo tipo, que han sido generados en la misma fase, mismo usuario, y que no se encuentren firmados, es decir, es una herramienta para propagar los cambios de un documento de tipo “múltiple” al resto de documentos del mismo tipo.

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento asociado al expediente. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String modo</i>	<i>Modo.</i> → “P” – Actualiza sólo el contenido de los párrafos de los documentos. → “F” – Actualiza sólo las firmas de los documentos. → “T” – Actualiza todo: párrafos y firmas.

2.4.1.4.1.3 Actualizar variables del documento del expediente

```
void actualizarVariablesDocumentoExp(
    TpoPK idDocExp,
    TrVariableDocExp[ ] variables,
    boolean elimVar) throws TrException
```

Permite actualizar las variables asociadas al documento que se indica en el parámetro *idDocExp*. El conjunto de variables se indica con el parámetro *variables* de los que sólo se tendrá en cuenta el atributo REFVARIABLE.

El parámetro *elimVar* puede tomar dos valores:

- **true:** Indica que se eliminen las variables que tuviese asociado el documento, con lo que a partir de entonces sólo tendrá asociadas las que se le pasen en el parámetro *variables*.
- **false:** Indica que no se eliminen las variables que tuviese asociado el documento, con lo que se añadirían las variables del parámetro *variables* a las ya existentes.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento del expediente al que se le van a asociar las variables. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TrVariableDocExp[] variables</i>	<i>Conjunto de variables para el documento a asociar al expediente. Sólo se usa el atributo REFVARIABLE.</i>
<i>boolean elimVar</i>	<i>Indica si se eliminan las variables que ya tenía el documento</i>

2.4.1.4.1.4 Modificar estado de un documento

```
void modificarEstadoDocumento(
    TpoPK idDocExpte,
    String estado,
    TpoDate fecha) throws TrException
```

Permite modificar el estado del documento que se indica como parámetro. Los valores válidos para el parámetro *estado* son: "R" (en realización), "E" (pendiente de firma), "F" (firmado), "D" (descartado), "T" (terminado), cualquier otro valor indicado no tiene efecto. Si el documento está en estado "F" no se puede modificar su estado.

El parámetro *fecha* permite dejar todas la firmas del documento a la fecha indicada en el caso de que el estado a modificar sea a "F". En este caso, para documentos generados con Report Server, también se genera el documento PDF correspondiente y se guarda en la base de datos.

Si se define un componente w@rdA en el sistema establecido en el TrAPIUI y el documento está guardado en el mismo w@rdA se actualiza su estado.

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento asociado al expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>String estado</i>	<i>Estado del documento</i> → "R" – En realización → "E" – Pendiente de firma → "F" – Firmado → "D" – Descartado → "T" – Terminado
	<i>Si el documento está en estado "F", no se podrá modificar su estado. Si el estado indicado es el mismo del documento, no se hace nada.</i>

TpoDate fecha

Fecha de la firma

Este parámetro permite dejar todas las firmas del documento a la fecha indicada en el caso de que el estado se vaya a modificar a "F". Si no se indica fecha, tomará por defecto la del sistema

void modificarEstadoDocumento(

**TpoPK idDocExpte,
String estado,
TpoDate fecha, TpoPK idUniOrg) throws TrException**

Entradas

TpoPK idDocExpte

*Identificador del documento asociado al expediente
Si el identificador es 'null', o no válido, se generará una excepción.*

String estado

*Estado del documento
→ "R" – En realización
→ "E" – Pendiente de firma
→ "F" – Firmado
→ "D" – Descartado
→ "T" – Terminado*

*Si el documento está en estado "F", no se podrá modificar su estado.
Si el estado indicado es el mismo del documento, no se hace nada.*

TpoDate fecha

Fecha de la firma

Este parámetro permite dejar todas las firmas del documento a la fecha indicada en el caso de que el estado se vaya a modificar a "F". Si no se indica fecha, tomará por defecto la del sistema

TpoPK idUniOrg

Identificador de la unidad orgánica

2.4.1.4.1.5 Asociar interesado a un documento

void asociarInteresadoDocumentoExpHijo(

TpoPK iidExpHijo) throws TrException

Permite asociar un interesado a un documento perteneciente a un expediente hijo.

Entradas

TpoPK iidExpHijo

Identificador del expediente hijo

2.4.1.4.1.6 Incorporar fichero a lista de documentos

```
void adjuntarFicheroMultipleDoc(List<TpoPK> docExps,
    InputStream fichero,
    String nombreFichero,
    String formato, String idWarda,
    String idWardaAnx,
    TpoPK idComp) throws TrException
```

Permite adjuntar un fichero físico a una lista de documentos incorporados que se indiquen como parámetro.

Entradas

<i>List<TpoPK > docExps</i>	Lista de documentos expedientes a incorporar fichero
<i>InputStream fichero</i>	Fichero a adjuntar
<i>String nombreFichero</i>	Nombre del fichero a adjuntar
<i>String formato</i>	Tipo "mime" del fichero
<i>String idWarda</i>	Identificador en w@rdA
<i>String idWardaAnx</i>	Identificador del anexo en w@rdA
<i>TpoPK idComp</i>	Identificador del componente w@rdA

2.4.1.4.1.7 Firmas documentos

TrFirmaDocumentoExpediente[] obtenerFirmasDocumento(
TpoPK idDocExpte,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener todos los datos de las firmas del documento que se indica como parámetro.

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento. Si el identificador es 'null' se devuelven todas las firmas de todos los documentos</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrFirmaDocumentoExpediente.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrFirmaDocumentoExpediente.</i>

Salida

TrFirmaDocumentoExpediente [] Array de objetos TrFirmaDocumentoExpediente.

TrFirmaDocumentoExpediente[] obtenerFirmasDocumento(
TpoPK idDocExpte,
boolean esExportar,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener todos los datos de las firmas del documento que se indica como parámetro.

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento. Si el identificador es 'null' se devuelven todas las firmas de todos los documentos</i>
<i>Boolean esExportar</i>	<i>Indica si la llamada al método se realiza para exportar el documento</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrFirmaDocumentoExpediente.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrFirmaDocumentoExpediente.</i>

Salida

TrFirmaDocumentoExpediente [] Array de objetos TrFirmaDocumentoExpediente.

2.4.1.4.1.8 Anular firma de un documento

```
void anularFirmaDocumento(
    TpoPK idDocExpte,
    String idPuestoTrab,
    TpoPK idUniOrg
    String usuarioPTUO ) throws TrException
```

Permite anular la firma de un documento que se indica como parámetro.

Sólo se permite anular la firma al usuario que la ordenó.

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String idPuestoTrab</i>	<i>Identificador del puesto de trabajo del firmante.</i>
<i>TpoPK idUniOrg</i>	<i>Identificador de la unidad orgánica del firmante.</i>
<i>String usuarioPTUO</i>	<i>Usuario que ocupa el puesto de trabajo.</i>

2.4.1.4.1.9 Anular firma de un interesado

```
void anularFirmaInteresadoDocumento(
    TpoPK idDocExpte,
    TpoPK idInteresado ) throws TrException
```

Permite anular la firma de un documento que se indica como parámetro.

Sólo se permite anular la firma al usuario que la ordenó.

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idInteresado</i>	<i>Identificador del interesado que realizó la firma. Si el identificador es 'null', o no válido, se generará una excepción</i>

2.4.1.4.1.10 Obtener interesados en un documento de un expediente

```
TrInteresadoDocumento[ ] obtenerInteresadosDocumento(
    TpoPK idExpediente,
    TpoPK idInteresado,
    TpoPK idRazonInt,
    TpoPK idRazonIntDocu,
    TpoPK idDocExp,
    ClausulaWhere where,
    ClausulaOrderBy orderBy) throws TrException
```


Permite obtener los datos de los interesados en los documentos de un expediente. Si se le pasa únicamente el parámetro *idExpediente* devolverá todos los interesados en todos los documentos de ese expediente.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idInteresado</i>	<i>Identificador del interesado</i> Opcional
<i>TpoPK idRazonInt</i>	<i>Identificador de la razón de interés del interesado del expediente</i> Opcional
<i>TpoPK idRazonIntDocu</i>	<i>Identificador de la razón de interés del interesado en el documento.</i> Opcional
<i>TpoPK idDocExp</i>	<i>Identificador del documento del expediente</i> Opcional
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrInteresadoDocumento.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrInteresadoDocumento.

Salida

TrInteresadoDocumento[] *Array de objetos TrInteresadoDocumento*

2.4.1.4.1.11 Insertar interesado en un documento de un expediente

```
void insertarInteresadoDocumento(
    TpoPK idExpediente,
    TpoPK idInteresado,
    TpoPK idRazonInt,
    TpoPK idRazonIntDocu,
    TpoPK idDocExp,
    String observaciones) throws TrException
```

Permite insertar un nuevo interesado en el documento. Se comprueba que existan todos los parámetros y además que el expediente no esté archivado.

Si se define un componente *w@rdA* en el sistema establecido en el *TrAPIUI* y el documento al que se le asigna el interesado está guardado en el mismo *w@rdA* se actualizan sus datos.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idInteresado</i>	<i>Identificador del interesado</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>

<i>TpoPK idRazonInt</i>	<i>Identificador de la razón de interés del interesado del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idRazonIntDocu</i>	<i>Identificador de la razón de interés del interesado en el documento. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idDocExp</i>	<i>Identificador del documento del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String observaciones</i>	<i>Observaciones para el nuevo interesado en el documento</i>

void insertarInteresadoDocumento(
TrInteresadoDocumento interesadoDocumento) throws TrException

Permite asociar a un interesado de un expediente, un documento por una razón de interés concreta.

Entradas

<i>TrInteresadoDocumento interesadoDocumento</i>	<i>Interesado del documento. Los atributos REFEXPEDIENTE, REFINTERESADO, REFRAZONINTEXP, REFRAZONINT y CONSENTIMIENTO son obligatorios.</i>
<i>String observaciones</i>	<i>Observaciones para el nuevo interesado en el documento</i>

2.4.1.4.1.12 Comprobar si el documento es editable

boolean comprobarEditablePorDoc(
TpoPK idDocExp) throws TrException

Comprueba si un documento es editable.

Entradas

<i>TpoPK idDocExp</i>	Identificador del documento.
-----------------------	------------------------------

2.4.1.4.1.13 Documentos del expediente

TrDocumentoExpediente[] obtenerDocumentosExpediente(
TpoPK idExpediente,
boolean soloUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve el conjunto de documentos que se han generado y/o incorporado al expediente dado. Además permite indicar si se filtra por el usuario establecido en el api y su perfil.

Si el identificador del expediente se pasa null, y no se usa ninguna cláusula where para filtrar, se devolverán todos los documento de todos los expedientes.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente</i> <i>Opcional</i>
<i>boolean soloUsuario</i>	<i>Indica si se filtra por el usuario del api y su perfil</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrDocumentoExpediente.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrDocumentoExpediente.

Salida

TrDocumentoExpediente []	<i>Array de objetos TrDocumentoExpediente.</i>
---------------------------	--

TrDocumentoExpediente[] obtenerDocumentosExpediente(

TpoPK idExpediente,
boolean soloUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy
ClausulaExclusion exclusiones) throws TrException

Devuelve el conjunto de documentos que se han generado y/o incorporado al expediente dado. Además permite indicar si se filtra por el usuario establecido en el api y su perfil.

Si el identificador del expediente se pasa null, y no se usa ninguna cláusula where para filtrar, se devolverán todos los documento de todos los expedientes.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente</i> <i>Opcional</i>
<i>boolean soloUsuario</i>	<i>Indica si se filtra por el usuario del api y su perfil</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrDocumentoExpediente.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrDocumentoExpediente.
<i>ClausulaExclusion exclusiones</i>	<i>Entidades a excluir de la consulta.</i> Consultar exclusiones posibles para TrDocumentoExpediente

Salida

TrDocumentoExpediente []	<i>Array de objetos TrDocumentoExpediente.</i>
---------------------------	--

TrDocumentoExpediente[] obtenerDocumentosExpediente(

TpoPK idExpediente,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve el conjunto de documentos que se han generado y/o incorporado al expediente dado. Este método está deprecado.

Si el identificador del expediente se pasa null, y no se usa ninguna cláusula where para filtrar, se devolverán todos los documento de todos los expedientes.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente</i> <i>Opcional</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrDocumentoExpediente.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrDocumentoExpediente.

Salida

TrDocumentoExpediente []	<i>Array de objetos TrDocumentoExpediente.</i>
---------------------------	--

2.4.1.4.1.14 Documentos del expediente por razón de interés

<p>TrDocumentoExpediente[] obtenerDocumentosExpedienteRazonInteres(<i>TpoPK idExpediente,</i> <i>String abreviaturaRazonInteres,</i> <i>boolean soloUsuario,</i> <i>ClausulaWhere where,</i> <i>ClausulaOrderBy orderBy,</i> <i>ClausulaExclusion exclusiones) throws TrException</i></p>

Devuelve el conjunto de documentos, pertenecientes al expediente dado, teniendo una razón de interés para una abreviatura dada.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente al que pertenecen los documentos</i>
<i>String abreviaturaRazonInteres</i>	<i>Abreviatura de la razón de interés que tiene el documento</i>
<i>Boolean soloUsuario</i>	<i>Indica si solo se obtienen los documentos del expediente del usuario</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrDocumentoExpediente.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrDocumentoExpediente.
<i>ClausulaExclusion exclusiones</i>	<i>Permite indicar qué datos no queremos obtener para agilizar la consulta a BBDD. Para ello debemos usar los campos de exclusión definidos en el tipo TrDocumentoExpediente.</i>

Salida

TrDocumentoExpediente []	<i>Array de objetos TrDocumentoExpediente.</i>
---------------------------	--

2.4.1.4.1.15 Comprobar si el documento es de tipo administrativo

boolean comprobarAdministrativoPorDoc(
String idDocExp) throws TrException

Comprueba si el documento es un documento electrónico administrativo válido o a un documento auxiliar que no forma parte del expediente.

Entradas

TpoPK idDocExp Identificador del documento expediente.

2.4.1.4.1.16 Comprobar si el tipo de documento es administrativo

boolean comprobarAdministrativoPorTipoDoc(
String idTipoDocExp) throws TrException

Comprueba si el tipo documento hace referencia a un documento electrónico administrativo válido o a un documento auxiliar que no forma parte del expediente.

Entradas

TpoPK idTipoDocExp Identificador del tipo de documento.

2.4.1.4.1.17 Incorporar documentos por referencia

void incorporarDocumentosPorReferencia(
TpoPK idExpte,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
List<String> referenciasCSV,
String motivo,
String nombreDoc,
TpoPK idExpteOriginal) throws TrException

Permite incorporar a un expediente una copia de un documento padre.

Entradas

TpoPK idExpte Identificador del expediente.
TpoPK idDocPer Identificador del documento permitido
TpoPK idDefProc Identificador del procedimiento
TpoDate fecha Fecha de alta

<i>String</i> presentado	Indica si está presentado o no. Puede ser 'S' o 'N'
<i>String</i> correcto	Indica si está presentado o no. Puede ser 'S' o 'N'
<i>List<String></i> referenciasCSV	Lista de CSV de los documentos referenciados
<i>String</i> motivo	Motivo de la incorporación
<i>String</i> nombreDoc	Nombre del documento
<i>TpoPK</i> idExpteOriginal	Identificador del expediente origen (padre)

2.4.1.4.1.18 Documentos permitidos en una fase

TrDocumentoPermitido[] obtenerDocumentosPermitidos(
TpoPK idFase,
TpoPK idDefProc,
TpoPK idExpediente,
TpoPK idExpFase
TpoDate fecha,
String condVisual,
boolean soloUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws *TrException*

Devuelve al usuario el conjunto de datos sobre los documentos que puede generar, incorporar, editar, etc., en una fase dada en un procedimiento también dado. Devuelve solo los documentos permitidos para un expediente concreto a una fecha dada, que cumplan las condiciones de visualización definidas en el documento.

Además permite indicar si se obtienen los que cumplan las condiciones de visualización o no y si se filtra por el usuario del api y su perfil.

Entradas

<i>TpoPK</i> idFase	Identificador de la fase. Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK</i> idDefProc	Identificador de la definición del procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK</i> idExpediente	Identificador del expediente. Opcional
<i>TpoPK</i> idExpFase	Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.
<i>TpoDate</i> fecha	Fecha Opcional. Si no se indica fecha, por defecto se tomará la del sistema.
<i>String</i> condVisual	Indica que documentos permitidos se obtienen → 'S' – Los que cumplan las condiciones → 'N' – Los que no cumplan las condiciones → 'T' – Todos
<i>boolean</i> soloUsuario	Indica si se filtra por el usuario establecido en el api y su perfil
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrDocumentoPermitido.

ClausulaOrderBy orderBy

Ordenación a aplicar.

Consultar campos posibles para TrDocumentoPermitido.

Salida

TrDocumentoPermitido []

Array de objetos TrDocumentoPermitido.

TrDocumentoPermitido[] obtenerDocumentosPermitidos(

TpoPK idFase,
TpoPK idDefProc,
TpoPK idExpediente,
TpoDate fecha,
String condVisual,
boolean soloUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve al usuario el conjunto de datos sobre los documentos que puede generar, incorporar, editar, etc, en una fase dada en un procedimiento también dado. Devuelve solo los documentos permitidos para un expediente concreto a una fecha dada, que cumplan las condiciones de visualización definidas en el documento. Además permite indicar si se cumplen las condiciones de visualización y filtrar por el usuario establecido en el api.

Entradas

TpoPK idFase

Identificador de la fase.

Si el identificador es 'null', o no válido, se generará una excepción.

TpoPK idDefProc

Identificador de la definición del procedimiento.

Si el identificador es 'null', o no válido, se generará una excepción.

TpoPK idExpediente

Identificador del expediente.

Opcional

TpoDate fecha

Fecha

Opcional. Si no se indica fecha, por defecto se tomará la del sistema.

String condVisual

Indica que documentos permitidos se obtienen

→ 'S' – Los que cumplan las condiciones

→ 'N' – Los que no cumplan las condiciones

→ 'T' – Todos

boolean soloUsuario

Indica si se filtra por el usuario establecido en el api y su perfil

ClausulaWhere where

Filtro a aplicar.

Consultar campos posibles para TrDocumentoPermitido.

ClausulaOrderBy orderBy

Ordenación a aplicar.

Consultar campos posibles para TrDocumentoPermitido.

Salida

TrDocumentoPermitido []

Array de objetos TrDocumentoPermitido.

TrDocumentoPermitido[] obtenerDocumentosPermitidos(

TpoPK idFase,
TpoPK idDefProc,
TpoPK idExpediente,

TpoDate fecha,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve al usuario el conjunto de datos sobre los documentos que puede generar, incorporar, editar, etc., en una fase dada en un procedimiento también dado. Devuelve solo los documentos permitidos para un expediente concreto a una fecha dada, que cumpla las condiciones de visualización definidas en el documento.

Entradas

<i>TpoPK idFase</i>	<i>Identificador de la fase. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i>
<i>TpoDate fecha</i>	<i>Opcional Fecha Opcional. Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrDocumentoPermitido.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrDocumentoPermitido.</i>

Salida

TrDocumentoPermitido [] *Array de objetos TrDocumentoPermitido.*

2.4.1.4.1.19 Eliminar un documento

void eliminarDocumento(
TpoPK idDocExp) throws TrException

Permite al usuario eliminar un documento de un expediente. Sólo se permite eliminar un documento generado/incorporado en la situación actual del expediente y que haya sido generado/incorporado por el mismo usuario que intenta eliminar y siempre que el documento se encuentre en realización.

Si se define un componente w@rdA en el sistema establecido en el TrAPIUI y el documento está guardado en el mismo w@rdA se elimina de éste.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento asociado al expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
-----------------------	---

2.4.1.4.1.20 Enviar documento expediente firma dinámica con anexo

void enviaDocumentoExpedienteFirmaDinamicaConAnexo(


```

Map<TpoPK, Boolean> m,
String aplicación,
String tipoDocu,
boolean notificaEmail,
boolean notificaAviso,
String[] estadosNotificacion,
TpoDate fechaInicio,
TpoDate fechaCaducidad,
String remitenteNombre,
String remitenteEmail,
String referencia,
String asunto,
BigDecimal prioridad,
String texto,
TrAccionDocumentoPortafirmas[] accionesPf,
String marcaVariable) throws TrException

```

Permite enviar un documento de un expediente con anexos a firmar en Port@Firmas sin necesidad de configurar previamente los firmantes y el tipo de firma.

Entradas

<i>Map<TpoPK, Boolean> m</i>	<i>Mapa que contiene los identificadores de los documentos expediente y si es anexo o no.</i>
<i>String aplicacion</i>	<i>Aplicación con la que se registra en Port@Firmas la petición</i>
<i>String tipoDocu</i>	<i>Tipo de documento, si no se envía coge el tipo del documento (opcional)</i>
<i>Boolean notificaEmail</i>	<i>Notificar por email (opcional)</i>
<i>Boolean notificaAviso</i>	<i>Notificar por avisador (opcional)</i>
<i>String[] estadosNotificacion</i>	<i>Estados en los que se quiere recibir notificaciones</i>
<i>TpoDate fechaInicio</i>	<i>Fecha de inicio de la firma</i>
<i>TpoDate fechaCaducidad</i>	<i>Fecha máxima de firma del documento</i>
<i>String remitenteNombre</i>	<i>Nombre del remitente de la petición</i>
<i>String remitenteEmail</i>	<i>Email del remitente de la petición</i>
<i>String referencia</i>	<i>Referencia o número de expediente</i>
<i>String asunto</i>	<i>Asunto</i>
<i>BigDecimal prioridad</i>	<i>Prioridad de la petición</i>
<i>String texto</i>	<i>Texto adjunto a la petición</i>
<i>TrAccionDocumentoPortafirmas[] accionesPf</i>	<i>Acciones para un cambio de estados del documento en Port@Firmas</i>
<i>String marcaVariable</i>	<i>Si se indica el parámetro marcaVariable distinto de cadena vacía y nulo, se comprueba que para los documentos pdf no existan variables sin sustituir ni con errores. Normalmente el valor de la variable se indicará "\$\$"</i>

2.4.1.4.1.21 Enviar documento expediente firma dinámica

```

void enviaDocumentoExpedienteFirmaDinamica(
    TpoPK[] arrayDocExp,
    String aplicación,
    String tipoDocu,
    boolean notificaEmail,
    boolean notificaAviso,
    String[] estadosNotificacion,
    TpoDate fechaInicio,
    TpoDate fechaCaducidad,
    String remitenteNombre,
    String remitenteEmail,
    String referencia,
    String asunto,
    BigDecimal prioridad,
    String texto,
    TrAccionDocumentoPortafirmas[] accionesPf,
    String marcaVariable) throws TrException

```

Método que envía varios documentos a Port@Firmas en una misma petición. Para ello, todos los documentos deben tener los mismos firmantes, tomándose como referencia el primer documento del array.

Entradas

<i>TpoPK[] m</i>	<i>Claves de los documentos</i>
<i>String aplicacion</i>	<i>Aplicación con la que se registra en Port@Firmas la petición</i>
<i>String tipoDocu</i>	<i>Tipo de documento, si no se envía coge el tipo del documento (opcional)</i>
<i>Boolean notificaEmail</i>	<i>Notificar por email (opcional)</i>
<i>Boolean notificaAviso</i>	<i>Notificar por avisador (opcional)</i>
<i>String[] estadosNotificacion</i>	<i>Estados en los que se quiere recibir notificaciones</i>
<i>TpoDate fechaInicio</i>	<i>Fecha de inicio de la firma</i>
<i>TpoDate fechaCaducidad</i>	<i>Fecha máxima de firma del documento</i>
<i>String remitenteNombre</i>	<i>Nombre del remitente de la petición</i>
<i>String remitenteEmail</i>	<i>Email del remitente de la petición</i>
<i>String referencia</i>	<i>Referencia o número de expediente</i>
<i>String asunto</i>	<i>Asunto</i>
<i>BigDecimal prioridad</i>	<i>Prioridad de la petición</i>
<i>String texto</i>	<i>Texto adjunto a la petición</i>
<i>TrAccionDocumentoPortafirmas[] accionesPf</i>	<i>Acciones para un cambio de estados del documento en Port@Firmas</i>
<i>String marcaVariable</i>	<i>Si se indica el parámetro marcaVariable distinto de cadena vacía y nulo, se comprueba que para los documentos pdf no existan variables sin sustituir ni con errores. Normalmente el valor de la variable se indicará "\$\$"</i>

2.4.1.4.1.22 Descarga documento

```
InputStream recuperarDocumentoExpediente(
    TpoPK idDocExp,
    TpoString formato,
    TpoString nombreFichero ) throws TrException
```

Permite descargar el contenido del documento que se indica como parámetro. Los parámetros *formato* y *nombreFichero* son de Entrada / Salida por si se quiere obtener el valor de dichos atributos.

Si se ha definido un componente *w@rdA* en el sistema establecido en el TrAPIUI se obtiene el documento de *w@rdA* siempre y cuando el *w@rdA* definido sea el mismo en el que se guardó el documento y su adjunto. Si no es el mismo *w@rdA* se devolverá null.

Si no se ha definido ningún *w@rdA* en el sistema se devolverá el documento almacenado en Trew@.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoString formato</i>	<i>Tipo "mime" del fichero. Si el documento se obtiene desde <i>w@rdA</i> este parámetro se devolverá vacío ya que en <i>w@rdA</i> no se recoge el formato del documento. Entrada/Salida.</i>
<i>TpoString nombreFichero</i>	<i>Nombre del fichero. Entrada/Salida.</i>

Salida

<i>InputStream</i>	<i>Fichero físico asociado al documento.</i>
--------------------	--

```
InputStream recuperarDocumentoExpediente(
    TpoPK idDocExp,
    TpoString formato,
    TpoString nombreFichero ) throws TrException
```

Permite descargar el contenido del documento que se indica como parámetro. Los parámetros *formato* y *nombreFichero* son de Entrada / Salida por si se quiere obtener el valor de dichos atributos.

Si se ha definido un componente *w@rdA* en el sistema establecido en el TrAPIUI se obtiene el documento de *w@rdA* siempre y cuando el *w@rdA* definido sea el mismo en el que se guardó el documento y su adjunto. Si no es el mismo *w@rdA* se devolverá null.

Si no se ha definido ningún *w@rdA* en el sistema se devolverá el documento almacenado en Trew@.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción</i>
-----------------------	---

<i>TpoString</i> formato	<i>Tipo "mime" del fichero. Si el documento se obtiene desde w@rdA este parámetro se devolverá vacío ya que en w@rdA no se recoge el formato del documento. Entrada/Salida.</i>
<i>TpoString</i> nombreFichero	<i>Nombre del fichero. Entrada/Salida.</i>

Salida

<i>InputStream</i>	Fichero físico asociado al documento.
--------------------	---------------------------------------

InputStream recuperarDocumentoExpediente(
TpoPK idDocExp,
TpoString formato,
TpoString nombreFichero,
String motivo) throws TrException

Devuelve un objeto *InputStream* cuyo contenido es el fichero del documento cuyo identificador es el que recibe como parámetro.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoString</i> formato	<i>Tipo "mime" del fichero. Si el documento se obtiene desde w@rdA este parámetro se devolverá vacío ya que en w@rdA no se recoge el formato del documento. Entrada/Salida.</i>
<i>TpoString</i> nombreFichero	<i>Nombre del fichero. Entrada/Salida.</i>
<i>TpoString</i> motivo	<i>Motivo de la consulta del documento</i>

Salida

<i>InputStream</i>	Fichero físico asociado al documento.
--------------------	---------------------------------------

2.4.1.4.1.23 Eliminar documentos múltiples

void eliminarDocumentosMultiples(
TpoPK idDocExpte) throws TrException

Permite eliminar los documentos definidos como “múltiples”, asociados a la misma fase, generados/incorporados por el mismo usuario y del mismo tipo que el pasado como parámetro siempre que no se encuentren firmados. Esta función también elimina el documento indicado (siempre en las mismas condiciones que el método *eliminarDocumento*).

Si se define un componente w@rdA en el sistema establecido en el TrAPIUI y el documento está guardado en el mismo w@rdA se elimina de éste.

Entradas

TpoPK idDocExpte *Identificador del documento asociado al expediente a eliminar.
Si el identificador es 'null', o no válido, se generará una excepción*

2.4.1.4.1.24 Enviar un documento a Port@firmas

```

void enviaDocumentoExpediente(
    TpoPK idDocExp,
    String aplicacion,
    String tipoDocu,
    boolean notificaEmail,
    boolean notificaAviso,
    String[] estadosNotificacion,
    TpoDate fechaInicio,
    TpoDate fechaCaducidad,
    String remitenteNombre,
    String remitenteEmail,
    String referencia,
    String asunto,
    BigDecimal prioridad,
    String texto) throws TrException

```

Método que envía uno o varios documentos a Port@firmas en una misma petición. Para ello todos los documentos deben tener los mismos firmantes, tomándose como referencia el primer documento del array. Permite indicar un array de acciones mediante el parámetro accionesPf para un cambio de estados del documento en Port@firmas.

Una de las excepciones más comunes que puede elevar este api es “Datos incongruentes en la petición” que puede estar causada por varios motivos:

- El sistema al que pertenece el documento no existe en port@firmas.
- El tipo de documento no existe en port@firmas.
- El usuario al que se le está enviando el documento no está dado de alta en port@firmas.
- La prioridad indicada no es correcta.
- El tablespace de port@firmas está lleno.
- Alguno de los campos indicados supera el límite de capacidad máximo. Los tamaños máximos permitidos por Port@firmas son:
 - remitenteNombre: 100 caracteres

- remitenteEmail: 50 caracteres
- referencia: 30 caracteres
- asunto 50: caracteres

Mediante el parámetro marcaVariable se puede controlar que no lleguen a Port@firmas documentos PDF con variables sin sustituir o con errores. Para ello se debe indicar la marca de variable usada en el documento, normalmente suele ser "\$\$". Si no se quiere hacer uso de esta comprobación, basta con pasar el parámetro a null.

Entradas

<i>TpoPK</i> idDocExp	Clave primaria del documento del expediente a enviar a port@firmas.
<i>String</i> aplicacion	Aplicación con la que se registra en port@firmas la petición
<i>String</i> tipoDocu	Tipo de documento (opcional). Si no se indica se envía la etiqueta del tipo de documento en Trew@.
<i>boolean</i> notificaEmail	Indica si se notifica por email (opcional)
<i>boolean</i> notificaAviso	Indica si se notifica por Avisador (opcional)
<i>String[]</i> estadosNotificacion	Estados en los que se quiere recibir notificaciones
<i>TpoDate</i> fechaInicio	Fecha de inicio de la firma
<i>TpoDate</i> fechaCaducidad	Fecha máxima de firma del documento
<i>String</i> remitenteNombre	Nombre del remitente de la petición
<i>String</i> remitenteEmail	email del remitente del petición
<i>String</i> referencia	Referencia o número de expediente
<i>String</i> asunto	Asunto
<i>BigDecimal</i> prioridad	Prioridad de la petición (1 a 5) o nulo valor por defecto 1 – Muy Alta 2 - Alta 3 – Media (Por defecto) 4 - Baja 5 – Muy Baja
<i>String</i> texto	Texto adjunto a la petición

<p>void enviaDocumentoExpediente(TpoPK[] arrayDocExp,</p>



String aplicacion,
String tipoDocu,
boolean notificaEmail,
boolean notificaAviso,
String[] estadosNotificacion,
TpoDate fechalnicio,
TpoDate fechaCaducidad,
String remitenteNombre,
String remitenteEmail,
String referencia,
String asunto,
BigDecimal prioridad,
String texto,
TrAccionDocumentoPortafirmas[] accionesPf) **throws TrException**

Entradas

<i>TpoPK</i> idDocExp	Clave primaria del documento del expediente a enviar a port@firmas.
<i>String</i> aplicacion	Aplicación con la que se registra en portafirmas la petición
<i>String</i> tipoDocu	Tipo de documento (opcional). Si no se indica se envía la etiqueta del tipo de documento en Trew@.
<i>boolean</i> notificaEmail	Indica si se notifica por email (opcional)
<i>boolean</i> notificaAviso	Indica si se notifica por Avisador (opcional)
<i>String[]</i> estadosNotificacion	Estados en los que se quiere recibir notificaciones
<i>TpoDate</i> fechalnicio	Fecha de inicio de la firma
<i>TpoDate</i> fechaCaducidad	Fecha máxima de firma del documento
<i>String</i> remitenteNombre	Nombre del remitente de la petición
<i>String</i> remitenteEmail	email del remitente del petición
<i>String</i> referencia	Referencia o número de expediente
<i>String</i> asunto	Asunto
<i>BigDecimal</i> prioridad	Prioridad de la petición (1 a 5) o nulo valor por defecto 1 – Muy Alta 2 - Alta 3 – Media (Por defecto) 4 - Baja 5 – Muy Baja
<i>String</i> texto	Texto adjunto a la petición

TrAccionDocumentoPortafirmas[]
accionesPf

Acciones para un cambio de estados del documento en Port@Firmas

```

void enviaDocumentoExpediente(
    TpoPK[] arrayDocExp,
    String aplicacion,
    String tipoDocu,
    boolean notificaEmail,
    boolean notificaAviso,
    String[] estadosNotificacion,
    TpoDate fechaInicio,
    TpoDate fechaCaducidad,
    String remitenteNombre,
    String remitenteEmail,
    String referencia,
    String asunto,
    BigDecimal prioridad,
    String texto,
    TrAccionDocumentoPortafirmas[] accionesPf,
    String marcaVariable) throws TrException

```

Entradas

<i>TpoPK</i> idDocExp	Clave primaria del documento del expediente a enviar a port@firmas.
<i>String</i> aplicacion	Aplicación con la que se registra en portafirmas la petición
<i>String</i> tipoDocu	Tipo de documento (opcional). Si no se indica se envía la etiqueta del tipo de documento en Trew@.
<i>boolean</i> notificaEmail	Indica si se notifica por email (opcional)
<i>boolean</i> notificaAviso	Indica si se notifica por Avisador (opcional)
<i>String[]</i> estadosNotificacion	Estados en los que se quiere recibir notificaciones
<i>TpoDate</i> fechaInicio	Fecha de inicio de la firma
<i>TpoDate</i> fechaCaducidad	Fecha máxima de firma del documento
<i>String</i> remitenteNombre	Nombre del remitente de la petición
<i>String</i> remitenteEmail	email del remitente del petición
<i>String</i> referencia	Referencia o número de expediente
<i>String</i> asunto	Asunto

<i>BigDecimal</i> prioridad	Prioridad de la petición (1 a 5) o nulo valor por defecto 1 – Muy Alta 2 - Alta 3 – Media (Por defecto) 4 - Baja 5 – Muy Baja
<i>String</i> texto	Texto adjunto a la petición
<i>TrAccionDocumentoPortafirmas[]</i> <i>accionesPf</i>	Acciones para un cambio de estados del documento en Port@Firmas
<i>String</i> marcaVariable	Si se indica el parámetro <i>marcaVariable</i> distinto de cadena vacía y nulo, se comprueba que para los documentos pdf no existan variables sin sustituir ni con errores. Normalmente el valor de la variable se indicará “\$\$”

2.4.1.4.1.25 Enviar documento a Port@Firmas con un documento anexo

```

void enviaDocumentoExpediente(
    Map<TpoPK, Boolean> m,
    String aplicación,
    String tipoDocu,
    boolean notificaEmail,
    boolean notificaAviso,
    String[] estadosNotificacion,
    TpoDate fechaInicio,
    TpoDate fechaCaducidad,
    String remitenteNombre,
    String remitenteEmail,
    String referencia,

```

String asunto,
BigDecimal prioridad,
String texto,
TrAccionDocumentoPortafirmas[] accionesPf,
String marcaVariable) **throws TrException**

Permite realizar envíos a Port@Firmas adjuntando un documento como anexo.

Entradas

<i>TpoPK</i> idDocExp	Clave primaria del documento del expediente a enviar a port@firmas.
<i>String</i> aplicacion	Aplicación con la que se registra en port@firmas la petición
<i>String</i> tipoDocu	Tipo de documento (opcional). Si no se indica se envía la etiqueta del tipo de documento en Trew@.
<i>boolean</i> notificaEmail	Indica si se notifica por email (opcional)
<i>boolean</i> notificaAviso	Indica si se notifica por Avisador (opcional)
<i>String[]</i> estadosNotificacion	Estados en los que se quiere recibir notificaciones
<i>TpoDate</i> fechaInicio	Fecha de inicio de la firma
<i>TpoDate</i> fechaCaducidad	Fecha máxima de firma del documento
<i>String</i> remitenteNombre	Nombre del remitente de la petición
<i>String</i> remitenteEmail	email del remitente del petición
<i>String</i> referencia	Referencia o número de expediente
<i>String</i> asunto	Asunto
<i>BigDecimal</i> prioridad	Prioridad de la petición (1 a 5) o nulo valor por defecto 1 – Muy Alta 2 - Alta 3 – Media (Por defecto) 4 - Baja 5 – Muy Baja
<i>String</i> texto	Texto adjunto a la petición
<i>TrAccionDocumentoPortafirmas[]</i> accionesPf	Acciones para un cambio de estados del documento en Port@Firmas
<i>String</i> marcaVariable	Si se indica el parámetro marcaVariable distinto de cadena vacía y nulo, se comprueba que para los documentos pdf no existan variables sin sustituir ni con errores. Normalmente el valor de la variable se indicará “\$\$”

2.4.1.4.1.26 Evaluar las condiciones asociadas a un documento

```

TrMensajeCondicionAccion[ ] evaluarCondicionesDocumento(
    TpoPK idExpediente,
    TpoPK idDocPer,
    TpoPK idDefProc,
    TpoPK idExpFase,
    String comprobar,
    TpoBoolean resultado,
    TpoDate fecha ) throws TrException

```

Permite al usuario evaluar el conjunto de condiciones asociadas a un documento para un expediente dado en un procedimiento también dado.

Entradas

<i>TpoPK idExpediente</i>	Identificador del expediente. Opcional. Sólo para condiciones en las que se quiera tener en cuenta este dato.
<i>TpoPK idDocPer</i>	Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idExpFase</i>	Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.
<i>String comprobar</i>	Indica cuáles son las condiciones que se quieren evaluar → "G" – Condiciones al generar el documento → "I" – Condiciones al incorporar el documento → "T" – Condiciones al generar e incorporar el documento → "V" – Condiciones al visualizar el documento
<i>TpoBoolean resultado</i>	Resultado. Entrada/Salida → "false" – Si alguna de las condiciones es obligatoria y no se cumple. → "true" – Si no existen condiciones obligatorias que no se cumplan.
<i>TpoDate fecha</i>	Fecha. Si no se indica fecha, tomará por defecto la del sistema.

Salida

TrMensajeCondicionAccion [] Array de objetos TrMensajeCondicionAccion.

```

TrMensajeCondicionAccion[ ] evaluarCondicionesDocumento(
    TpoPK idExpediente,
    TpoPK idDocPer,
    TpoPK idDefProc,
    String comprobar,
    TpoBoolean resultado,
    TpoDate fecha) throws TrException

```

Permite al usuario evaluar el conjunto de condiciones asociadas a un documento para un expediente dado en un procedimiento también dado.

Entradas

<i>TpoPK idExpediente</i>	Identificador del expediente. Opcional. Sólo para condiciones en las que se quiera tener en cuenta este dato.
<i>TpoPK idDocPer</i>	Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.
<i>String comprobar</i>	Indica cuáles son las condiciones que se quieren evaluar → "G" – Condiciones al generar el documento → "I" – Condiciones al incorporar el documento → "T" – Condiciones al generar e incorporar el documento → "V" – Condiciones al visualizar el documento
<i>TpoBoolean resultado</i>	Resultado. Entrada/Salida → "false" – Si alguna de las condiciones es obligatoria y no se cumple. → "true" – Si no existen condiciones obligatorias que no se cumplan.
<i>TpoDate fecha</i>	Fecha. Si no se indica fecha, tomará por defecto la del sistema.

Salida

TrMensajeCondicionAccion [] Array de objetos TrMensajeCondicionAccion.

2.4.1.4.1.27 establece parámetros documento

```
void estableceParametrosDocumento(
    TpoPK idDocExpte,
    TrValorParametro[] datosParametros) throws TrException
```

Establece los parámetros para un documento pasado como parámetro.

Entradas

<i>TpoPK idDocExp</i>	Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción
<i>TrValorParametro[] datosParametros</i>	Array de valores para parámetros. Si se pasa como VALOR de un parámetro la cadena \$REF_DOEX\$ se establecerá por defecto el Identificador del documento del expediente

2.4.1.4.1.28 Exportar documento externo ENI

String exportarDocExternoEni (
 TpoPK idDocExpte,
 OutputStream xml,
 String motivo) throws TrException

Exporta un documento expediente perteneciente a un sistema externo con formato xml especificado por el Esquema Nacional de Interoperabilidad (ENI).

Entrada

<i>TpoPK idDocExpte</i>	Identificador de documento expediente
<i>OutputStream xml</i>	Fichero xml del documento electrónico en formato ENI
String motivo	Motivo para obtener los documentos de repositorios externos en caso que aplique.

Salida

<i>String</i>	Identificador del documento ENI generado
---------------	--

2.4.1.4.1.29 Exportar documento externo ENI administrador

String exportarDocExternoEniAdmon (
 TpoPK idDocExpte,
 OutputStream xml,
 String motivo,
 boolean admon) throws TrException

Exporta un documento de un repositorio externo perteneciente a un sistema en una cadena con formato xml especificado por el Esquema Nacional de Interoperabilidad (ENI). Si el documento expediente no tiene identificador ENI asociado, se generará un nuevo identificador para el documento, en caso contrario no se modificará el identificador existente.

Entrada

<i>TpoPK idDocExpte</i>	Identificador de documento expediente
<i>OutputStream xml</i>	Fichero xml del documento electrónico en formato ENI

String motivo	Motivo para obtener los documentos de repositorios externos en caso que aplique.
Boolean admon	Valor que determina si es administrador o no.

Salida

String	Identificador del documento ENI generado
--------	--

2.4.1.4.1.30 Comprobar si existe documento externo en expediente

boolean existeDocExternoEnExpediente(TpoPK idExpte) throws TrException

Permite comprobar si el expediente contiene al menos un documento existente en un repositorio externo.

Entradas

<i>TpoPK idExpte</i>	<i>Identificador del expediente.</i>
----------------------	--------------------------------------

2.4.1.4.1.31 Obtener la fecha de firma de un documento

String obtenerFechaFirmaDocumento(TpoPK idTipoDoc) throws TrException
Deprecado

Mediante esta función se obtiene si para un tipo de documento aparece o no la fecha en el pie de firma. Devuelve 'S' en caso de que si aparezca y 'N' en caso contrario.

Entrada

<i>TpoPK idTipoDoc</i>	Identificador del tipo de documento
------------------------	-------------------------------------

Salida

String

Fecha de la firma del documento

2.4.1.4.1.32 Comprueba si existe documento borrado

boolean existeDocumentoExpBorrado(
String csv) throws *TrException*

Comprueba si el documento ha sido eliminado a través del csv pasado por parámetro.

Entradas

String csv

Csv del documento.

2.4.1.4.1.33 Exportar documento ENI

String exportarDocumentoENI (
TpoPK idDocExpte,
OutputStream xml) throws *TrException*

Exporta un documento expediente perteneciente a un sistema una cadena con formato xml especificado por el Esquema Nacional de Interoperabilidad (ENI). Si el documento expediente no tiene identificador ENI asociado, se generará un nuevo identificador el documento, en caso contrario no se modificará el identificador existente.

Entrada

TpoPK idDocExpte

Identificador de documento expediente

OutputStream xml

Fichero xml del documento electrónico en formato ENI

Salida

String

Identificador del documento ENI generado

2.4.1.4.1.34 Realizar expedientes sin firmarlos

List<Object> firmaDelegadaDocumento (
byte[] documento) throws *TrException*

Permite delegar la firma de un documento a Trew@. Requiere la configuración de la conexión remota por EJB

Entrada

byte[] documento

Array de bytes del contenido del documento

Salida

List<Object>

Devuelve una lista de Object que contiene dos elementos: identificador de transacción de firma y array del documento firmado

2.4.1.4.1.35 Firmar dinámicamente documento

<p>void firmarDinamicamenteDocumento(TpoPK idDocExp, TpoDate fecha, String idPuestoTrab, TpoPK idUniOrg, String usuarioPTUO, String tipoFir, String editable, String txtPie, String txtFdo) throws TrException</p>	<i>Deprecado</i>
---	------------------

Permite firmar un documento que se pasa como parámetro por un firmante que no se ha definido previamente como firmante del documento, por lo que cualquier firmante podrá firmar el documento. Se debe indicar el puesto, unidad orgánica del firmante, así como si es principal, sustituto o delegado de otro, debido a que, por ejemplo, un mismo firmante puede ser principal y sustituto de otro a la vez.

Se podrá firmar siempre que se tengan permisos para ello en el tramitador.

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento que se va a firmar. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoDate fecha</i>	<i>Fecha de la firma. Si no se indica nada, por defecto se toma la del sistema.</i>
<i>String idPuestoTrab</i>	<i>Identificador del puesto de trabajo del firmante.</i>
<i>TpoPK idUniOrg</i>	<i>Identificador de la unidad orgánica del firmante.</i>
<i>String usuarioPTUO</i>	<i>Usuario que ocupa el puesto de trabajo.</i>
<i>String tipFir</i>	<i>Indica si el firmante indicado es principal 'P', sustituto 'S' ó delegado 'D'. Si es null se tomará por defecto el valor 'P'.</i>
<i>String editable</i>	<i>Indica si el pie de firma es editable o no. → "S" – Sí es editable → "N" – No es editable</i>
<i>String txtPie</i>	<i>Texto de pie de firma.</i>
<i>String txtFdo</i>	<i>Texto firmado.</i>

Ya que varios usuarios pueden ocupar el mismo puesto, indicando el usuario que ocupa el puesto, se dice qué usuario es realmente el que firma.

Si el tipo de documento a firmar es en "cascada", el orden de firmantes vendrá dado por el orden en el que disitintos firmantes hayan realizado una firma sobre el documento.


```

void firmarDinamicamenteDocumento(
    TpoPK idDocExp,
    TpoDate fecha,
    String idPuestoTrab,
    TpoPK idUniOrg,
    String usuarioPTUO,
    String tipoFir,
    String editable,
    String txtPie,
    String txtFdo,
    String tipoFirmaEni) throws TrException

```

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento que se va a firmar. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoDate fecha</i>	<i>Fecha de la firma. Si no se indica nada, por defecto se toma la del sistema.</i>
<i>String idPuestoTrab</i>	<i>Identificador del puesto de trabajo del firmante.</i>
<i>TpoPK idUniOrg</i>	<i>Identificador de la unidad orgánica del firmante.</i>
<i>String usuarioPTUO</i>	<i>Usuario que ocupa el puesto de trabajo.</i>
<i>String tipFir</i>	<i>Indica si el firmante indicado es principal 'P', sustituto 'S' ó delegado 'D'. Si es null se tomará por defecto el valor 'P'.</i>
<i>String editable</i>	<i>Indica si el pie de firma es editable o no. → "S" – Sí es editable → "N" – No es editable</i>
<i>String txtPie</i>	<i>Texto de pie de firma.</i>
<i>String txtFdo</i>	<i>Texto firmado.</i>
<i>String tipoFirmaEni</i>	<i>Indica el tipo de firma ENI. → TF01- CSV → TF02 - XAdES internally detached signature. → TF03 - XAdES enveloped signature. → TF04 - CADES detached/explicit signature. → TF05 - CADES attached/implicit signature. → TF06 - PAdES.</i>

2.4.1.4.1.36 Firmantes tipo documentos

```

TrFirmanteTipoDocumento[ ] obtenerFirmantesTipoDocumento(
    TpoPK idTipoDoc,
    TpoDate fecha,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException

```

Permite obtener los firmantes definidos en el sistema y vigentes a la fecha que se pasa como parámetro para el tipo de documento que se indica.

Como un mismo documento puede tener distintos firmantes dependiendo de la Unidad Orgánica donde estemos tratando el documento, la lista de firmantes se devolverá en función de la Unidad Orgánica del usuario que hace la llamada al método, para ello se filtra por el tipo de Unidad Orgánica y la provincia.

Si se quiere desactivar el filtro por el tipo de Unidad Orgánica y provincia se debe hacer uso del api `setFiltroFirmantes(false)`. Además se pueden realizar filtros y ordenaciones mediante los parámetros `where` y `orderBy`.

Entradas

<i>TpoPK</i> idTipoDoc	<i>Identificador del tipo de documento.</i>
<i>TpoDate</i> fecha	<i>Fecha</i> Si no se indica fecha de alta, por defecto se tomará la del sistema.
<i>ClausulaWhere</i> where	<i>Filtro a aplicar.</i> Consultar campos posibles para TrFirmanteTipoDocumento.
<i>ClausulaOrderBy</i> orderBy	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrFirmanteTipoDocumento.

Salida

TrFirmanteTipoDocumento [] *Array de objetos TrFirmanteTipoDocumento.*

TrFirmanteTipoDocumento [] obtenerFirmantesTipoDocumento(
TpoPK idTipoDoc,
TpoDate fecha) throws TrException

Entradas

<i>TpoPK</i> idTipoDoc	<i>Identificador del tipo de documento.</i>
<i>TpoDate</i> fecha	<i>Fecha</i> Si no se indica fecha de alta, por defecto se tomará la del sistema.

Salida

TrFirmanteTipoDocumento [] *Array de objetos TrFirmanteTipoDocumento.*

TrFirmanteTipoDocumento [] obtenerFirmantesTipoDocumento(
TpoPK idTipoDoc,
TpoDate fecha,
ClausulaWhere where,
ClausulaOrderBy orderBy,
TpoPK idUniOrg) throws TrException

Entradas

<i>TpoPK idTipoDoc</i>	<i>Identificador del tipo de documento.</i>
<i>TpoDate fecha</i>	<i>Fecha</i> Si no se indica fecha de alta, por defecto se tomará la del sistema.
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrFirmanteTipoDocumento.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrFirmanteTipoDocumento.
<i>TpoPK idUniOrg</i>	<i>Identificador de la unidad organizativa</i>

Salida

TrFirmanteTipoDocumento [] *Array de objetos TrFirmanteTipoDocumento.*

2.4.1.4.1.37 Fecha firma tipo documento

<p>String fechaFirmaTipoDocumento (<i>TpoPK idTipoDoc) throws TrException</i></p>
--

Mediante esta función se obtiene si para un tipo de documento aparece o no la fecha en el pie de firma. Devuelve "S" en caso de que sí aparezca y "N" en caso contrario

Entradas

<i>TpoPK idTipoDoc</i>	<i>Identificador del tipo de documento.</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>
------------------------	---

Salida

<i>String</i>	<i>Indica si existe o no fecha en el pie de firma.</i> → " S " – Sí existe fecha en el pie de firma → " N " – No existe fecha en el pie de firma
---------------	--

2.4.1.4.1.38 Anotar un documento como generado

```

TrMensajeCondicionAccion[ ] generarDocumento(
    TpoPK idExpediente,
    TpoPK idDocPer,
    TpoPK idDefProc,
    TpoDate fecha,
    TpoDate fechaLimite,
    TpoPK expFase,
    String observaciones,
    TrValorParametro[ ] datosParametros,
    TpoPK idDocExpte ,
    TrPlantillaTipoDocumento plantilla ) throws TrException

```

Permite al usuario generar un documento dado para un expediente en un procedimiento también suministrado. El método devuelve un conjunto de mensajes/avisos derivados de las condiciones establecidas para la generación del documento. Además se puede indicar la fecha límite para el documento del expediente.

Si se ha definido un componente w@rdA en el sistema establecido en el TrAPIUI se guardará el documento generado en w@rdA.

Si se define en el tipo de documento la propiedad "Fusionar variables" a "Sí" se realizará la sustitución de las variables existentes en los párrafos cuya propiedad "Fusionar variables" también esté a "Sí" o en el documento OpenOffice de las variables asociadas al tipo de documento (ver más detalles en Configuración generación documentos OpenOffice).

Si la plantilla se pasa como parámetro anula la consulta de plantilla del tipo de documento generándose dicha plantilla, de lo contrario generará la implementada en la lógica de la función asociada a la consulta en caso que este asociada y válida para el tipo de documento.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido Opcional.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoDate fecha</i>	<i>Fecha en la cual el documento queda anotado para el expediente como "generado". Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite para el documento del expediente</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String observaciones</i>	<i>Observaciones de tramitación a la fase. Esta cadena no puede tener una longitud superior a 250 caracteres.</i>
<i>TrValorParametro[] datosParametros</i>	<i>Conjunto de parámetros a utilizar para la sustitución de variables contenidas en los párrafos que forman el documento. Si se pasa como VALOR de un parámetro la cadena \$REF_DOEX\$ se establecerá por defecto el Identificador del documento del expediente</i>

TpoPK idDocExpte

Identificador único que TREW@ le da al documento generado para ese expediente.
Entrada/Salida.

TrPlantillaTipoDocumento

La plantilla del tipo de documento.
Si el identificador es 'null', se generará la primera del listado.

Salida

TrMensajeCondicionAccion []

Array de objetos TrMensajeCondicionAccion.

TrMensajeCondicionAccion[] generarDocumento(

TpoPK idExpediente,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
String observaciones,
TrValorParametro[] datosParametros,
TpoPK idDocExp) throws TrException

Deprecado

Entradas

TpoPK idExpediente

Identificador del expediente.
Si el identificador es 'null', o no válido, se generará una excepción.

TpoPK idDocPer

Identificador del documento permitido
Opcional.

TpoPK idDefProc

Identificador de la definición de procedimiento.
Si el identificador es 'null', o no válido, se generará una excepción.

TpoDate fecha

Fecha en la cual el documento queda anotado para el expediente como "generado".

String observaciones

Si no se indica fecha, por defecto se tomará la del sistema.

Observaciones de tramitación a la fase.

Esta cadena no puede tener una longitud superior a 250 caracteres.

TrValorParametro[] datosParametros

Conjunto de parámetros a utilizar para la sustitución de variables contenidas en los párrafos que forman el documento.

Si se pasa como VALOR de un parámetro la cadena \$REF_DOEX\$ se establecerá por defecto el Identificador del documento del expediente

TpoPK idDocExp

Identificador único que TREW@ le da al documento generado para ese expediente.

Salida

TrMensajeCondicionAccion []

Array de objetos TrMensajeCondicionAccion.

TrMensajeCondicionAccion[] generarDocumento(

TpoPK idExpediente,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
TpoDate fechaLimite,
String observaciones,

TrValorParametro[] datosParametros,
TpoPK idDocExp) throws TrException

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido Opcional.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoDate fecha</i>	<i>Fecha en la cual el documento queda anotado para el expediente como "generado". Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite para el documento del expediente</i>
<i>String observaciones</i>	<i>Observaciones de tramitación a la fase. Esta cadena no puede tener una longitud superior a 250 caracteres.</i>
<i>TrValorParametro[] datosParametros</i>	<i>Conjunto de parámetros a utilizar para la sustitución de variables contenidas en los párrafos que forman el documento. Si se pasa como VALOR de un parámetro la cadena \$REF_DOEX\$ se establecerá por defecto el Identificador del documento del expediente</i>
<i>TpoPK idDocExp</i>	<i>Identificador único que TREW@ le da al documento generado para ese expediente.</i>

Salida

<i>TrMensajeCondicionAccion []</i>	<i>Array de objetos TrMensajeCondicionAccion.</i>
-------------------------------------	---

TrMensajeCondicionAccion[] generarDocumento(

*TpoPK idExpediente,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
TrValorParametro[] datosParametros,
TpoPK idDocExp) throws TrException*

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido Opcional.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>

<i>TpoDate</i> fecha	<i>Fecha en la cual el documento queda anotado para el expediente como "generado".</i>
<i>TpoDate</i> fechaLimite	<i>Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>TpoPK</i> expFase	<i>Fecha límite para el documento del expediente</i>
<i>String</i> observaciones	<i>Identificador de la instancia de la fase.</i>
<i>TrValorParametro[]</i> datosParametros	<i>Si el identificador es 'null', el valor será la última instancia.</i>
<i>TpoPK</i> idDocExp	<i>Observaciones de tramitación a la fase.</i>
	<i>Esta cadena no puede tener una longitud superior a 250 caracteres.</i>
	<i>Conjunto de parámetros a utilizar para la sustitución de variables contenidas en los párrafos que forman el documento.</i>
	<i>Si se pasa como VALOR de un parámetro la cadena \$REF_DOEX\$ se establecerá por defecto el identificador del documento del expediente</i>
	<i>Identificador único que TREW@ le da al documento generado para ese expediente.</i>

Salida

TrMensajeCondicionAccion [] Array de objetos *TrMensajeCondicionAccion*.

<p>TrMensajeCondicionAccion[] generarDocumento(</p> <p> <i>TpoPK</i> idExpediente, <i>TpoPK</i> idDocPer, <i>TpoPK</i> idDefProc, <i>TpoDate</i> fecha, <i>TpoDate</i> fechaLimite, <i>TpoPK</i> expFase, String observaciones, <i>TrValorParametro[]</i> datosParametros, <i>TpoPK</i> idDocExpte , <i>TrPlantillaTipoDocumento</i> plantilla, String numDoc, <i>TpoPK</i> idUniOrg) throws <i>TrException</i></p> <p align="right"><i>Deprecado</i></p>
--

Entradas

<i>TpoPK</i> idExpediente	<i>Identificador del expediente.</i>
<i>TpoPK</i> idDocPer	<i>Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK</i> idDefProc	<i>Identificador del documento permitido</i>
<i>TpoDate</i> fecha	<i>Opcional.</i>
<i>TpoDate</i> fechaLimite	<i>Identificador de la definición de procedimiento.</i>
<i>TpoPK</i> expFase	<i>Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>String</i> observaciones	<i>Fecha en la cual el documento queda anotado para el expediente como "generado".</i>
	<i>Si no se indica fecha, por defecto se tomará la del sistema.</i>
	<i>Fecha límite para el documento del expediente</i>
	<i>Identificador de la instancia de la fase.</i>
	<i>Si el identificador es 'null', el valor será la última instancia.</i>
	<i>Observaciones de tramitación a la fase.</i>
	<i>Esta cadena no puede tener una longitud superior a 250 caracteres.</i>

<i>TrValorParametro[] datosParametros</i>	<i>Conjunto de parámetros a utilizar para la sustitución de variables contenidas en los párrafos que forman el documento. Si se pasa como VALOR de un parámetro la cadena \$REF_DOEX\$ se establecerá por defecto el Identificador del documento del expediente</i>
<i>TpoPK idDocExpte</i>	<i>Identificador único que TREW@ le da al documento generado para ese expediente. Entrada/Salida.</i>
<i>TrPlantillaTipoDocumento</i>	<i>La plantilla del tipo de documento. Si el identificador es 'null', se generará la primera del listado.</i>
<i>String numDoc</i>	<i>Número del documento</i>
<i>TpoPK idUniOrg</i>	<i>Identificador del organismo de captura</i>

Salida

<i>TrMensajeCondicionAccion []</i>	<i>Array de objetos TrMensajeCondicionAccion.</i>
-------------------------------------	---

Si existen condiciones obligatorias definidas sobre el documento y alguna no se cumple, no se generará a no ser que el usuario tenga permisos de administración.

<p>TrMensajeCondicionAccion[] generarDocumentoMultiple(TpoPK idExpediente, TpoPK idDocPer, TpoPK idDefProc, TpoDate fecha, TpoDate fechaLimite, TpoPK expFase, String observaciones, TrValorParametro[] datosParametros, TrDocMultiple[] IdDocMultiples, TrPlantillaTipoDocumento idPlantilla) throws TrException</p>
--

Permite al usuario generar múltiples documentos para un expediente en un procedimiento, dependiendo del valor del atributo *comprobarBloqueo* se tendrá en cuenta si el expediente se encuentra bloqueado o no. Los documentos deben ser del mismo tipo. Devuelve un conjunto de mensajes/avisos derivados de las condiciones y acciones establecidas para la generación del documento.

La generación de documentos múltiples devuelve en el parámetro de Entrada/Salida *idDocMultiples* un array de *TrDocMultiple* compuesto por dos atributos *TpoPK*:

- REFAUX: Identificador auxiliar, para uso de aplicaciones externas.
- REFDOCEXP: Identificador del documento generado en Trew@.

Si se ha definido un componente *w@rdA* en el sistema establecido en el TrAPIUI se guardará el documento generado en *w@rdA*.

Si la plantilla se pasa como parámetro anula la consulta de plantilla del tipo de documento generándose dicha plantilla, de lo contrario generará la implementada en la lógica de la función asociada a la consulta en caso que este asociada y válida para el tipo de documento.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido Opcional.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoDate fecha</i>	<i>Fecha en la cual el documento queda anotado para el expediente como "generado". Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite para el documento del expediente</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String observaciones</i>	<i>Observaciones de tramitación a la fase. Esta cadena no puede tener una longitud superior a 250 caracteres.</i>
<i>TrValorParametro[] datosParametros</i>	<i>Conjunto de parámetros a utilizar para la sustitución de variables contenidas en los párrafos que forman el documento. Si se pasa como VALOR de un parámetro la cadena \$REF_DOEX\$ se establecerá por defecto el Identificador del documento del expediente</i>
<i>TrDocMultiple[] IdDocMultiples</i>	<i>Array de identificadores únicos que TREW@ le da al documento/s generado/s.</i>
<i>TrPlantillaTipoDocumento plantilla</i>	<i>Entrada/Salida. La plantilla del tipo de documento</i>

Salida

<i>TrMensajeCondicionAccion []</i>	<i>Array de objetos TrMensajeCondicionAccion.</i>
-------------------------------------	---

TrMensajeCondicionAccion[] generarDocumentoMultiple(

*TpoPK idExpediente,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
TrValorParametro[] datosParametros,
TrDocMultiple[] idDocMultiples) throws TrException*

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
---------------------------	---

<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido</i>
<i>TpoPK idDefProc</i>	<i>Opcional.</i> <i>Identificador de la definición de procedimiento.</i> <i>Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoDate fecha</i>	<i>Fecha en la cual el documento queda anotado para el expediente como "generado".</i> <i>Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite para el documento del expediente</i>
<i>String observaciones</i>	<i>Observaciones de tramitación a la fase.</i> <i>Esta cadena no puede tener una longitud superior a 250 caracteres.</i>
<i>TrValorParametro[] datosParametros</i>	<i>Conjunto de parámetros a utilizar para la sustitución de variables contenidas en los párrafos que forman el documento.</i> <i>Si se pasa como VALOR de un parámetro la cadena \$REF_DOEX\$ se establecerá por defecto el Identificador del documento del expediente</i>
<i>TrDocMultiple[] idDocMultiples</i>	<i>Array de identificadores únicos que TREW@ le da al documento/s generado/s.</i> <i>Entrada/Salida.</i>

Salida

<i>TrMensajeCondicionAccion []</i>	<i>Array de objetos TrMensajeCondicionAccion.</i>
-------------------------------------	---

TrMensajeCondicionAccion[] generarDocumentoMultiple(

TpoPK idExpediente,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
TrValorParametro[] datosParametros,
TrDocMultiple[] idDocMultiples) throws TrException

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> <i>Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido</i>
<i>TpoPK idDefProc</i>	<i>Opcional.</i> <i>Identificador de la definición de procedimiento.</i> <i>Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoDate fecha</i>	<i>Fecha en la cual el documento queda anotado para el expediente como "generado".</i> <i>Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite para el documento del expediente</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase.</i> <i>Si el identificador es 'null', el valor será la última instancia.</i>

String observaciones *Observaciones de tramitación a la fase.*
Esta cadena no puede tener una longitud superior a 250 caracteres.

TrValorParametro[] datosParametros *Conjunto de parámetros a utilizar para la sustitución de variables contenidas en los párrafos que forman el documento.*
Si se pasa como VALOR de un parámetro la cadena \$REF_DOEX\$ se establecerá por defecto el Identificador del documento del expediente

TrDocMultiple[] idDocMultiples *Array de identificadores únicos que TREW@ le da al documento/s generado/s.*
Entrada/Salida.

Salida

TrMensajeCondicionAccion [] Array de objetos TrMensajeCondicionAccion.

2.4.1.4.1.39 Exportar expediente ENI con documentos

<p>File exportarExpedienteEniConDocumentos(TpoPK idExpte) throws TrException <i>Deprecado</i></p>

Genera un zip con la información del expediente electrónico y su índice firmado, así como todos los documentos electrónicos firmados del expediente. Devuelve un fichero zip que empaqueta el xml de expediente y sus documentos. Los documentos existentes en el tramitador serán actualizados con el identificador de documento generado para el índice, al igual que el expediente.

Entradas

TpoPK idExpte *Identificador del expediente.*
Si el identificador es 'null', o no válido, se generará una excepción.

Salida

File Fichero ZIP con los xml de expediente y sus documentos en formato ENI.

<p>File exportarExpedienteEniConDocumentos(TpoPK idExpte, boolean serializacionAvanzada) throws TrException <i>Deprecado</i></p>
--

Entradas

TpoPK idExpte *Identificador del expediente.*
Si el identificador es 'null', o no válido, se generará una excepción.

Boolean serializacionAvanzada *Indica si se utilizará para la serialización una librería compatible con Jboss 7 y JDK superiores a la 5*

Salida

	Junta de Andalucía Agencia Digital de Andalucía	Trew@ Guía de referencia de JTrApi
---	---	---------------------------------------

File

Fichero ZIP con los xml de expediente y sus documentos en formato ENI.

2.4.1.4.1.40 Incorporar documento externo en múltiples expedientes

List<TpoPK> incorporarDocExternoMultiplesExpedientes(
List<TpoPK> exps,
TpoPK idDocPer,
TpoPK idDefProc,
TpoPK idFase,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
String observaciones,
String csv,
String motivo,
String nombreDoc) throws TrException

Permite al usuario incorporar un documento externo dado para varios expedientes en un procedimiento también suministrado, en caso de que algun expediente esté bloqueado se podrá incorporar el documento siempre y cuando el atributo *comprobarBloqueo* sea false

. Sólo se persistirán los metadatos del documento permaneciendo el contenido del mismo custodiado en el repositorio externo. El método devuelve una lista con los identificadores de los documentos incorporados.

Entradas

<i>List<TpoPK> exps</i>	<i>Lista de expedientes</i> Si la lista es 'null', o no válido y vacía, se generará una excepción.
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido.</i> Si el identificador es 'null' se generará una excepción.
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idFase</i>	<i>Identificador de la fase</i>
<i>TpoDate fecha</i>	<i>Fecha de incorporación del documento.</i> Es la fecha en la cual el documento queda anotado para el expediente como "incorporado". Si no se indica fecha, por defecto se tomará la del sistema.
<i>String presentado</i>	Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado
<i>String correcto</i>	Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto
<i>TpoDate fechaLimite</i>	Fecha límite de presentación o de corrección del documento.
<i>TpoPK expFase</i>	Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.
<i>String observaciones</i>	Observaciones para anotar a la hora de incorporar el documento.
<i>String csv</i>	Csv del documento original.
<i>String motivo</i>	Motivo para realizar la consulta al repositorio externo.
<i>String nombreDoc</i>	Nombre del documento.

Salida

List<TpoPK> docs

Lista de objetos TpoPK con los identificadores de documentos generados.

2.4.1.4.1.41 Incorporar documento externo no definido otro expediente

void incorporarDocExternoNoDefOtroExpediente(

TpoPK idExpediente,
TpoPK idTipoDoc,
TpoPK idFase,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
TpoPK idDocExpOrig,
String csv
String motivo,
String nombreDoc) throws TrException

Permite al usuario incorporar a un expediente una copia de un documento externo existente en el mismo u otro expediente como documento no definido. Sólo se persistirán los metadatos del documento permaneciendo el contenido del mismo custodiado en el repositorio externo.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente donde incorporamos el documento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idTipoDoc</i>	<i>Identificador del tipo de documento que se quiere incorporar. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idFase</i>	<i>Identificador de la fase del expediente a la que se quiere asociar el documento incorporado. Debe ser una de las fases en las que se encuentre el expediente.</i>
<i>TpoDate fecha</i>	<i>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado". Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>String presentado</i>	<i>Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado</i>
<i>String correcto</i>	<i>Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite de presentación o de corrección del documento.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String observaciones</i>	<i>Observaciones para anotar a la hora de incorporar el documento.</i>
<i>TpoPK idDocExpOrig</i>	<i>Identificador del documento expediente original. Documento a copiar.</i>
<i>TpoPK idDocExpte</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente.</i>
<i>String csv</i>	<i>Entrada/Salida. Csv del documento original.</i>



String motivo

Motivo para realizar la consulta al repositorio externo.

String nombreDoc

Nombre del documento.

2.4.1.4.1.42 Incorporar documento no definido externo

void incorporarDocumentoNoDefinidoExterno(

```
TpoPK idExpte,  
TpoPK idTipDoc,  
TpoPK idFase,  
TpoDate fecha,  
String presentado,  
String correcto,  
TpoDate fechaLimite,  
TpoPK expFase,  
String observaciones,  
TpoPK idDocExpte,  
String csv  
String motivo,  
String nombreDoc ) throws TrException
```

Permite incorporar un documento externo dado para un expediente no recogido en la definición del procedimiento. Sólo se persistirán los metadatos del documento permaneciendo el contenido del mismo custodiado en el repositorio externo.

Entradas

TpoPK idExpediente

Identificador del expediente.

Si el identificador es 'null', o no válido, se generará una excepción.

TpoPK idTipDoc

Identificador del tipo de documento que se quiere incorporar.

Si el identificador es 'null', o no válido, se generará una excepción.

TpoPK idFase

Identificador de la fase del expediente a la que se quiere asociar el documento incorporado.

Debe ser una de las fases en las que se encuentre el expediente.

Si el identificador es 'null', o no válido, se generará una excepción.

TpoDate fecha

Fecha de incorporación del documento.

Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".

String presentado

Si no se indica fecha, por defecto se tomará la del sistema.

Indica si el documento se ha presentado o no.

→ "**S**" – El documento se ha presentado

→ "**N**" – El documento no se ha presentado

String correcto

Indica si el documento que se ha presentado es correcto o no.

→ "**S**" – El documento es correcto

→ "**N**" – El documento no es correcto

TpoDate fechaLimite

Fecha límite de presentación o de corrección del documento.

TpoPK expFase

Identificador de la instancia de la fase.

Si el identificador es 'null', el valor será la última instancia.

String observaciones

Observaciones para anotar a la hora de incorporar el documento.

<i>TpoPK</i> idDocExpte	Identificador único que TREW@ le da al documento incorporado para ese expediente.
<i>String</i> csv	Entrada/Salida. Csv del documento original.
<i>String</i> motivo	Motivo para realizar la consulta al repositorio externo.
<i>String</i> nombreDoc	Nombre del documento.

2.4.1.4.1.43(w) Anotar un documento como incorporado en varios expedientes

List<TpoPK> incorporarDocumentoMultiplesExpedientes(

List<TpoPK> exps,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
TpoPK idEstElabo,
String numDocOrig) throws TrException

Permite al usuario incorporar un documento dado para varios expedientes en un procedimiento también suministrado. El método devuelve una lista con los identificadores de los documentos incorporados.

Si se ha definido un componente w@rdA en el sistema establecido en el TrAPIUI también se guardarán los documentos incorporados en w@rdA.

Entradas

<i>List<TpoPK></i> exps	<i>Lista de expedientes</i> Si la lista es 'null', o no válido y vacía, se generará una excepción.
<i>TpoPK</i> idDocPer	<i>Identificador del documento permitido.</i> Si el identificador es 'null' se generará una excepción.
<i>TpoPK</i> idDefProc	<i>Identificador de la definición de procedimiento.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoDate</i> fecha	<i>Fecha de incorporación del documento.</i> Es la fecha en la cual el documento queda anotado para el expediente como "incorporado". Si no se indica fecha, por defecto se tomará la del sistema.
<i>String</i> presentado	Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado
<i>String</i> correcto	Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto
<i>TpoDate</i> fechaLimite	Fecha límite de presentación o de corrección del documento.
<i>TpoPK</i> expFase	Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.
<i>String</i> observaciones	Observaciones para anotar a la hora de incorporar el documento.

TpoPK idEstElab

Identificador del estado de elaboración del documento

String numDocOrig

Csv del documento original

Salida

List<TpoPK> docs

Lista de objetos TpoPK con los identificadores de documentos generados.

List<TpoPK> incorporarDocumentoMultiplesExpedientes(

List<TpoPK> exps,
TpoPK idDocPer,
TpoPK idDefProc,
TpoPK idFase,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
String observaciones) throws TrException

Entradas

List<TpoPK> exps

Lista de expedientes

Si la lista es 'null', o no válido y vacía, se generará una excepción.

TpoPK idDocPer

Identificador del documento permitido.

Si el identificador es 'null' se generará una excepción.

TpoPK idDefProc

Identificador de la definición de procedimiento.

Si el identificador es 'null', o no válido, se generará una excepción.

TpoPK idFase

Identificador de la fase donde se encuentran los expedientes

TpoDate fecha

Fecha de incorporación del documento.

Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".

String presentado

Si no se indica fecha, por defecto se tomará la del sistema.

Indica si el documento se ha presentado o no.

→ "S" – El documento se ha presentado

→ "N" – El documento no se ha presentado

String correcto

Indica si el documento que se ha presentado es correcto o no.

→ "S" – El documento es correcto

→ "N" – El documento no es correcto

TpoDate fechaLimite

Fecha límite de presentación o de corrección del documento.

String observaciones

Observaciones para anotar a la hora de incorporar el documento.

Salida

List<TpoPK> docs

Lista de objetos TpoPK con los identificadores de documentos generados.

List<TpoPK> incorporarDocumentoMultiplesExpedientes(

List<TpoPK> exps,
TpoPK idDocPer,
TpoPK idDefProc,
TpoPK idFase,

<p>TpoDate fecha, String presentado, String correcto, TpoDate fechaLimite, String observaciones) throws TrException</p>	<p><i>Deprecado</i></p>
---	-------------------------

Entradas

<i>List<TpoPK> exps</i>	<p><i>Lista de expedientes</i> Si la lista es 'null', o no válido y vacía, se generará una excepción.</p>
<i>TpoPK idDocPer</i>	<p><i>Identificador del documento permitido.</i> Si el identificador es 'null' se generará una excepción.</p>
<i>TpoPK idDefProc</i>	<p><i>Identificador de la definición de procedimiento.</i> Si el identificador es 'null', o no válido, se generará una excepción.</p>
<i>TpoPK idFase</i>	<p>Identificador de la fase donde se encuentran los expedientes</p>
<i>TpoDate fecha</i>	<p><i>Fecha de incorporación del documento.</i> Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".</p>
<i>String presentado</i>	<p>Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no.</p>
<i>String correcto</i>	<p>Indica si el documento que se ha presentado es correcto o no.</p>
<i>TpoDate fechaLimite</i>	<p>→ "S" – El documento se ha presentado → "N" – El documento no se ha presentado</p>
<i>String observaciones</i>	<p>Indica si el documento es correcto → "S" – El documento es correcto → "N" – El documento no es correcto</p>
<i>String numDoc</i>	<p>Fecha límite de presentación o de corrección del documento.</p>
<i>TpoPK idUniOrg</i>	<p>Observaciones para anotar a la hora de incorporar el documento.</p>
<i>TpoPK idEstElab</i>	<p>Número del documento</p>
<i>String numDocOrig</i>	<p>Identificador del organismo de captura</p>
	<p>Identificador de estado de elaboración del documento ENI</p>
	<p>Número de documento origen en caso de que el documento ENI sea una copia con cambio de formato o una copia parcial</p>

Salida

<i>List<TpoPK> docs</i>	<p>Lista de objetos TpoPK con los identificadores de documentos generados.</p>
-------------------------------	--

2.4.1.4.1.44 Incorporar documento no definido de otro expediente

```

void incorporarDocumentoNoDefOtroExpediente(
    TpoPK idExpte,
    TpoPK idTipDoc,
    TpoPK idFase,
    TpoDate fecha,
    String presentado,
    String correcto,
    TpoDate fechaLimite,
    TpoPK expFase,
    String observaciones,
    TpoPK idDocExpOrig,
    TpoPK idDocExpte,
    TpoPK idTipEni,
    TpoPK idEstElabo,
    String numDocOrig) throws TrException
  
```

Permite al usuario incorporar a un expediente una copia de un documento existente en el mismo u otro expediente como documento no definido.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente donde incorporamos el documento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idTipoDoc</i>	<i>Identificador del tipo de documento que se quiere incorporar. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idFase</i>	<i>Identificador de la fase del expediente a la que se quiere asociar el documento incorporado. Debe ser una de las fases en las que se encuentre el expediente.</i>
<i>TpoDate fecha</i>	<i>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado". Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>String presentado</i>	<i>Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado</i>
<i>String correcto</i>	<i>Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite de presentación o de corrección del documento.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String observaciones</i>	<i>Observaciones para anotar a la hora de incorporar el documento.</i>
<i>TpoPK idDocExpOrig</i>	<i>Identificador del documento expediente original. Documento a copiar.</i>
<i>TpoPK idDocExpte</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.</i>
<i>TpoPK idTipEni</i>	<i>Identificador del tipo documental eni</i>
<i>TpoPK idEstElab</i>	<i>Identificador del estado de elaboración del documento</i>
<i>String numDocOrig</i>	<i>Csv del documento original</i>



void incorporarDocumentoNoDefOtroExpediente(

TpoPK idExpte,
TpoPK idTipDoc,
TpoPK idFase,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
TpoPK idDocExpOrig,
TpoPK idDocExpte) throws TrException

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente donde incorporamos el documento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idTipoDoc</i>	<i>Identificador del tipo de documento que se quiere incorporar. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idFase</i>	<i>Identificador de la fase del expediente a la que se quiere asociar el documento incorporado.</i>
<i>TpoDate fecha</i>	<i>Debe ser una de las fases en las que se encuentre el expediente. Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".</i>
<i>String presentado</i>	<i>Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado</i>
<i>String correcto</i>	<i>Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite de presentación o de corrección del documento.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String observaciones</i>	<i>Observaciones para anotar a la hora de incorporar el documento.</i>
<i>TpoPK idDocExpOrig</i>	<i>Identificador del documento expediente original. Documento a copiar.</i>
<i>TpoPK idDocExpte</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.</i>

void incorporarDocumentoNoDefOtroExpediente(

TpoPK idExpte,
TpoPK idTipDoc,
TpoPK idFase,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,

<p>TpoPK idDocExpOrig, TpoPK idDocExppte, TpoPK idTipEni, String numDoc, TpoPK idUniOrg, TpoPK idEstElab, String numDocOrig) throws TrException</p> <p style="text-align: right;"><i>Deprecado</i></p>
--

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente donde incorporamos el documento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idTipoDoc</i>	<i>Identificador del tipo de documento que se quiere incorporar. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idFase</i>	<i>Identificador de la fase del expediente a la que se quiere asociar el documento incorporado. Debe ser una de las fases en las que se encuentre el expediente.</i>
<i>TpoDate fecha</i>	<i>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado". Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>String presentado</i>	<i>Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado</i>
<i>String correcto</i>	<i>Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite de presentación o de corrección del documento.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String observaciones</i>	<i>Observaciones para anotar a la hora de incorporar el documento.</i>
<i>TpoPK idDocExpOrig</i>	<i>Identificador del documento expediente original. Documento a copiar.</i>
<i>TpoPK idDocExppte</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.</i>
<i>TpoPK idTipEni</i>	<i>Identificador del tipo documental eni</i>
<i>String numDoc</i>	<i>Número del documento</i>
<i>TpoPK idUnirOrg</i>	<i>Identificador de estado de elaboración del documento ENI</i>
<i>TpoPK idEstElab</i>	<i>Identificador de estado de elaboración del documento ENI</i>
<i>String numDocOrig</i>	<i>Número de documento origen en caso de que el documento ENI sea una copia con cambio de formato o una copia parcial.</i>

2.4.1.4.1.45 Incorporar documento de otro expediente

<p>TrMensajeCondicionAccion[] incorporarDocumentoOtroExpediente(</p>

TpoPK idExpte,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
TpoPK idDocExpOrig,
TpoPK idDocExp,
TpoPK idEstElab,
String numDocOrig) throws TrException

Permite al usuario incorporar a un expediente una copia de un documento existente en el mismo u otro expediente. El método devuelve un conjunto de mensajes/avisos derivados de las condiciones establecidas para la incorporación del documento.

Entradas

<i>TpoPK idExpte</i>	<i>Identificador del expediente donde incorporamos el documento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido al cual pertenece el documento a incorporar.</i>
<i>TpoPK idDefProc</i>	<i>Si el identificador es 'null', o no válido, se generará una excepción. Identificador de la definición del procedimiento al que pertenece el expediente.</i>
<i>TpoDate fecha</i>	<i>Si el identificador es 'null', o no válido, se generará una excepción. Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".</i>
<i>String presentado</i>	<i>Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado</i>
<i>String correcto</i>	<i>Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite de presentación o de corrección del documento.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String observaciones</i>	<i>Observaciones para anotar a la hora de incorporar el documento.</i>
<i>TpoPK idDocExpOrig</i>	<i>Identificador del documento expediente original. Documento a copiar.</i>
<i>TpoPK idDocExpte</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.</i>
<i>TpoPK idEstaElab</i>	<i>Identificador del estado de elaboración del documento</i>
<i>String numDocOrig</i>	<i>Csv del documento original</i>

TrMensajeCondicionAccion[] incorporarDocumentoOtroExpediente(

TpoPK idExpte,
TpoPK idDocPer,

<p>TpoPK idDefProc, TpoDate fecha, String presentado, String correcto, TpoDate fechaLimite, TpoPK expFase, String observaciones, TpoPK idDocExpOrig, TpoPK idDocExp, String numDoc, TpoPK idUniOrg, TpoPK idEstElab, String numDocOrig) throws TrException</p>	<p align="right"><i>Deprecado</i></p>
--	---------------------------------------

Entradas

<i>TpoPK idExpte</i>	<i>Identificador del expediente donde incorporamos el documento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido al cual pertenece el documento a incorporar. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento al que pertenece el expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoDate fecha</i>	<i>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".</i>
<i>String presentado</i>	<i>Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado</i>
<i>String correcto</i>	<i>Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite de presentación o de corrección del documento.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String observaciones</i>	<i>Observaciones para anotar a la hora de incorporar el documento.</i>
<i>TpoPK idDocExpOrig</i>	<i>Identificador del documento expediente original. Documento a copiar.</i>
<i>TpoPK idDocExp</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.</i>
<i>String numDoc</i>	<i>Número del documento</i>
<i>TpoPK idUniOrg</i>	<i>Identificador del organismo de captura</i>
<i>TpoPK idEstaElab</i>	<i>Identificador del estado de elaboración del documento</i>
<i>String numDocOrig</i>	<i>Csv del documento original</i>

TrMensajeCondicionAccion[] incorporarDocumentoOtroExpediente(

TpoPK idExpte,
TpoPK idDocPer,

<p>TpoPK idDefProc, TpoDate fecha, String presentado, String correcto, TpoDate fechaLimite, TpoPK expFase, String observaciones, TpoPK idDocExpOrig, TpoPK idDocExp) throws TrException</p> <p style="text-align: right;"><i>Deprecado</i></p>
--

Entradas

<i>TpoPK idExpte</i>	<i>Identificador del expediente donde incorporamos el documento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido al cual pertenece el documento a incorporar. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento al que pertenece el expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoDate fecha</i>	<i>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".</i>
<i>String presentado</i>	<i>Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado</i>
<i>String correcto</i>	<i>Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite de presentación o de corrección del documento.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String observaciones</i>	<i>Observaciones para anotar a la hora de incorporar el documento.</i>
<i>TpoPK idDocExpOrig</i>	<i>Identificador del documento expediente original. Documento a copiar.</i>
<i>TpoPK idDocExp</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.</i>

2.4.1.4.1.46 Intercambiar orden párrafos

<p>void intercambiarOrdenParrafosDocumento(TpoPK idParrafolni, TpoPK idParrafofin) throws TrException</p>

Permite intercambiar el orden de los párrafos de un documento. Se comprueba que los párrafos existan y pertenezcan al mismo documento.

Entradas

<i>TpoPK</i> idParrafoIni	Identificador del párrafo a intercambiar el orden.
<i>TpoPK</i> idParrafoFin	Identificador del otro párrafo a intercambiar el orden.

2.4.1.4.1.47 Sustituir variables documentos

boolean sustituirVariablesDocumento(*TpoPK* idDocExp) throws TrException

Sustituye las variables y actualiza los párrafos de los documentos indicados. Se sustituyen las variables del documento indicados *TpoPK idDocExp*.

Además se puede utilizar este api para sustituir las variables de los documentos de generación OpenOffice bean. Para ello las variables deben estar asociadas al tipo de documento para que al generarse el documento se guarden junto a éste.

Para los documentos cuyo modo de generación sea OpenOffice bean se debe configurar (separador filas, columnas marcas variable, prefijos variables tablas e imágenes) (ver más detalles en Configuración generación documentos OpenOffice).

Entradas

<i>TpoPK</i> idDocExp	Identificador del documento que se quiere sustituir las variables.
-----------------------	--

Salida Boolean true Si es satisfactorio

boolean sustituirVariablesDocumento(
TpoPK[] idDocsExp,
boolean multiple) throws TrException *Deprecado*

Entradas

<i>TpoPK[]</i> idDocsExp	Identificadores de los documentos que se quiere sustituir las variables.
--------------------------	--

Boolean multiple	Indica si el documento es múltiple que se sustituyan las variables para todos los documentos "hermanos" del indicado.
------------------	---

2.4.1.4.1.48 Generación de documentos Java PDF

byte[] generarDocumentoPDF(*TpoPK[]* idDocsExp,
boolean multiple,
boolean pieFirma,
String versionados,
boolean informePortafirmas) throws TrException

Genera un documento Java Pdf. Permite indicar si el documento es múltiple o no, en caso de que se indice múltiple a true, se añadiran al Pdf los documentos hermanos (documentos del mismo tipo, en la misma fase y por el mismo usuario) del indicado. También permite indicar si se quiere incluir los datos del pie de firma. El parámetro versionados indica si se añaden al pdf los documentos versionados, se pueden indicar los valores: "S": Sólo los versionados, "N": No se añaden los versionados ó "T": Todos. El parámetro informePortafirmas indica si se quiere obtener el documento con el informe de firma de Port@firmas, para ello se deben pasar a true los parámetros *pieFirma* e *informePortafirmas* y además el documento tiene que tener código hash Port@firmas y su estado debe ser firmado.

Para obtener el informe de firma de Port@firmas (*pieFirma* = true e *informePortafirmas* = true) se tiene que definir la constante del sistema (sistema establecido o sistema por defecto) `URL_INFORME_PFIRMA` cuyo valor será la url que devuelve el documento firmado con el informe de firma. Por ejemplo:

`http://servidor:puerto/pfirma/informeFirma/?docHASH=#HASHDOC#&petHASH=#HASHPET#`

En el parámetro `#HASHDOC#` el api indicará automáticamente el código hash del documento y en el parámetro `#HASHPET#` el api indicará el código has de la petición de Port@firmas.

Si se quiere obtener el documento con el informe de firma de Trew@ se deben pasar el parámetro *pieFirma* = true e *informePortafirmas* = false. Existe la posibilidad de obtener el código de transacción cifrado mediante una clave 3DES indicada en el componente de firma que se pasara al api *incluirFirmaDigital* al guardar los datos de la firma digital. Para ello se debe dar de alta el dato componente `CLAVE_3DES` en el componente indicado cuyo valor sea una clave 3DES.

Además se puede especificar un texto en el que se incluya la forma de verificar la veracidad del documento. Para ello se debe dar de alta el dato componente `TEXTO_VERIFICA` en el componente indicado cuyo valor sea el nombre de una constante del sistema o sistema por defecto que será la que tenga el texto que aparecerá en el informe de firma. Si se quiere que en este texto aparezca el código de transacción se debe incluir la cadena `$IDTRANSACCION$`.

Esta api permite indicar un array de identificadores de documentos del expediente para generarlos todos en un mismo documento pdf.

Antes de usar la generación Java PDF se debe de haber establecido la configuración para dicha generación mediante el api `establecerConfiguracionGeneracionPdf` y debe existir la constante de sistema `URL_REPORT_JAVA` con la url de los recursos para la generación pdf (platilla xml, por ejemplo `DOC_GENERAL.xml`, imágenes, fuentes, etc.). Por ejemplo `http://localhost:8080/AgendaTrewa/editor/config/`

Además se puede utilizar este api para generar un documento PDF a partir de un documento cuyo modo de generación sea OpenOffice bean. Para ello se debe configurar el servidor de Open Office (ver más detalles en Configuración generación documentos OpenOffice).

Entrada

<i>tpoPK</i> idDocsExp	Identificador/es del documento/s.
<i>boolean</i> multiple	Indica si el documento es múltiple y se generan en pdf los documentos hermanos.
<i>boolean</i> pieFirma	Indica si a los documentos se le añade el pie de firma.

String versionados Indica si genera los documentos versionados.
Sus valores pueden ser:

- **"S"** – Sí, sólo saca los documentos versionados, cuyo estado es "V"
- **"N"** – No, sólo saca los documentos no versionados cuyo estado es distinto de "V"
- **"T"** – Todos

Boolean informePortafirmas Indica si se quiere obtener el informe de firma de Trew@ (false) o de Port@firmas (true), solo es válido para documentos firmados con firma digital. Si se quiere obtener se debe indicar el parámetro pieFirma a true y haber configurado la constante URL_INFORME_PFIRMA del sistema del documento o en el sistema por defecto.

Salida

byte[] Array de bytes con el/los documento/s en Pdf

byte[] generarDocumentoPDF(TpoPK[] idDocsExp,
boolean multiple,
boolean pieFirma,
String versionados,
boolean informePortafirmas
OutputStream finalDocumenPdf) throws TrException

Entrada

TpoPK idDocsExp Identificador/es del documento/s.

boolean multiple Indica si el documento es múltiple y se generan en pdf los documentos hermanos.

boolean pieFirma Indica si a los documentos se le añade el pie de firma.

String versionados Indica si genera los documentos versionados.
Sus valores pueden ser:

- **"S"** – Sí, sólo saca los documentos versionados, cuyo estado es "V"
- **"N"** – No, sólo saca los documentos no versionados cuyo estado es distinto de "V"
- **"T"** – Todos

Boolean informePortafirmas Indica si se quiere obtener el informe de firma de Trew@ (false) o de Port@firmas (true), solo es válido para documentos firmados con firma digital. Si se quiere obtener se debe indicar el parámetro pieFirma a true y haber configurado la constante URL_INFORME_PFIRMA del sistema del documento o en el sistema por defecto.

OutputStream
finalDocumentPdf Documento pdf generado

Salida

byte[] Array de bytes con el/los documento/s en Pdf

byte[] generarDocumentoPDF(TpoPK[] idDocsExp,
boolean multiple,
boolean pieFirma,
String versionados) throws TrException

Entrada

<i>TpoPK</i> idDocsExp	Identificador/es del documento/s.
<i>boolean</i> multiple	Indica si el documento es múltiple y se generan en pdf los documentos hermanos.
<i>boolean</i> pieFirma	Indica si a los documentos se le añade el pie de firma.
<i>String</i> versionados	Indica si genera los documentos versionados. Sus valores pueden ser: <ul style="list-style-type: none"> - "S" – Sí, sólo saca los documentos versionados, cuyo estado es "V" - "N" – No, sólo saca los documentos no versionados cuyo estado es distinto de "V" - "T" – Todos

Salida

<i>byte[]</i>	Array de bytes con el/los documento/s en Pdf
----------------	--

2.4.1.4.1.49Valores de variables de un documento

Map obtenerValoresVariablesDocumento(TpoPK idDocExpte boolean incluirVarPárrafos) throws TrException

Devuelve un mapa con todas las variables del documento y su correspondiente valor, para ello las variables deben estar asociadas al documento. Si se indica el parámetro *incluirVarPárrafos* a true se parsean los párrafos para el caso de los documentos con estructura de párrafos o el documento OpenOffice en el caso de documentos OpenOffice bean (ver más detalles en Configuración generación documentos OpenOffice)., y se obtienen las variables que existan en el texto. Para ello es necesario indicar el parámetro *incluirVarPárrafos* a true. Al usar este api existe la posibilidad de que la variable devuelva un java.util.Map con el valor de varias variables. La estructura del mapa será un String en la key con el nombre de la variable y un String en el value con el valor de la variable.

Entradas

<i>TpoPK</i> idDocExp	<i>Identificador del documento.</i> Si el identificador es 'null', o no válido, se generará una excepción
<i>boolean</i> incluirVarPárrafos	Indica si se quieren parsear los párrafos del documento para obtener las variables existentes en el contenido que no estén asociadas al documento. Si se indica <i>true</i> se realizará el parseo de todos los párrafos, si se indica <i>false</i> no se realizará éste parseo y se obtendrán sólo las variables asociadas al documento.

Salida

<i>Map</i>	Devuelve un mapa cuya key es un String con el nombre de la variable y el value es un String con el valor de la variable. Si no existen variables asociadas al documento ni en párrafos (en caso de que se indica <i>true</i> en <i>incluirVarPárrafos</i>) devolverá <i>null</i> .
------------	---

2.4.1.4.1.50Valores de variables

Map<String, String> obtenerValoresVariables(

ArrayList<TpoPK> idsVariables,
TpoPK idDocExpte) throws TrException

Devuelve un mapa clave valor con el nombre y el valor de la variable utilizando los parámetros asignados al documento.

Entradas

ArrayList<TpoPK> idsVariables	Identificadores de las variables. Si la lista de identificadores es 'null', o no válido, se generará una excepción
TpoPK idDocExpte	Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción

Salida

Map<String, String>	Mapa con los nombres y valores de las variables
---------------------	---

2.4.1.4.1.51 Obtener DocumentoEni externo

DocumentoEni ObtenerDocumentoEniExterno(
String csv,
String motivo) throws TrException

Permite obtener un documento custodiado en un repositorio externo indicando el csv y el motivo como parámetros.

Entradas

String csv	Csv del documento custodiado en repositorio externo.
String motivo	Motivo para realizar la consulta al repositorio externo.

Salida

DocumentoEni	Objeto DocumentoEni
--------------	---------------------

2.4.1.4.1.52(w) Anotar un documento como incorporado

TrMensajeCondicionAccion[] incorporarDocumento(
TpoPK idExpediente,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
TpoPK idDocExp,

TpoPK idEstElab,
String numDocOrg) throws TrException

Permite al usuario incorporar un documento dado para un expediente en un procedimiento también suministrado, en caso de que el expediente bloqueado se podrá incorporar el documento siempre y cuando el atributo *comprobarBloqueo* sea false. El método devuelve un conjunto de mensajes/avisos derivados de las condiciones establecidas para la incorporación del documento.

Si se ha definido un componente w@rdA en el sistema establecido en el TrAPIUI también se guardará el documento incorporado en w@rdA.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido.</i> Si el identificador es 'null' se generará una excepción.
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoDate fecha</i>	<i>Fecha de incorporación del documento.</i> <i>Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".</i>
<i>String presentado</i>	Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado
<i>String correcto</i>	Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto
<i>TpoDate fechaLimite</i>	Fecha límite de presentación o de corrección del documento.
<i>TpoPK expFase</i>	Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.
<i>String observaciones</i>	Observaciones para anotar a la hora de incorporar el documento.
<i>TpoPK idDocExp</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente.</i> <i>Entrada/Salida.</i>
<i>TpoPk idEstElab</i>	<i>Identificador del estado de elaboración del documento</i>
<i>String numDocOrig</i>	<i>Csv del documento original</i>

Salida

<i>TrMensajeCondicionAccion []</i>	Array de objetos TrMensajeCondicionAccion.
-------------------------------------	--

TrMensajeCondicionAccion[] incorporarDocumento(
TpoPK idExpediente,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
String presentado,

String correcto,
TpoDate fechaLimite,
String observaciones,
TpoPK idDocExp) throws TrException

Entradas

<i>TpoPK</i> idExpediente	<i>Identificador del expediente.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK</i> idDocPer	<i>Identificador del documento permitido.</i> Si el identificador es 'null' se generará una excepción.
<i>TpoPK</i> idDefProc	<i>Identificador de la definición de procedimiento.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoDate</i> fecha	<i>Fecha de incorporación del documento.</i> Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".
<i>String</i> presentado	Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado
<i>String</i> correcto	Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto
<i>TpoDate</i> fechaLimite	Fecha límite de presentación o de corrección del documento.
<i>String</i> observaciones	Observaciones para anotar a la hora de incorporar el documento.
<i>TpoPK idDocExp</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente.</i> <i>Entrada/Salida.</i>

Salida

<i>TrMensajeCondicionAccion</i> []	Array de objetos <i>TrMensajeCondicionAccion</i> .
-------------------------------------	--

TrMensajeCondicionAccion[] incorporarDocumento(
TpoPK idExpediente,
TpoPK idDocPer,
TpoPK idDefProc,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
TpoPK idDocExp) throws TrException

Entradas

<i>TpoPK</i> idExpediente	<i>Identificador del expediente.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK</i> idDocPer	<i>Identificador del documento permitido.</i> Si el identificador es 'null' se generará una excepción.

<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoDate fecha</i>	<i>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".</i>
<i>String presentado</i>	<i>Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no.</i> → "S" – El documento se ha presentado → "N" – El documento no se ha presentado
<i>String correcto</i>	<i>Indica si el documento que se ha presentado es correcto o no.</i> → "S" – El documento es correcto → "N" – El documento no es correcto
<i>TpoDate fechaLimite</i>	<i>Fecha límite de presentación o de corrección del documento.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase</i>
<i>String observaciones</i>	<i>Observaciones para anotar a la hora de incorporar el documento.</i>
<i>TpoPK idDocExp</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.</i>

Salida

<i>TrMensajeCondicionAccion []</i>	<i>Array de objetos TrMensajeCondicionAccion.</i>
-------------------------------------	---

<p>TrMensajeCondicionAccion[] incorporarDocumento(<i>TpoPK idExpediente,</i> <i>TpoPK idDocPer,</i> <i>TpoPK idDefProc,</i> <i>TpoDate fecha,</i> <i>String presentado,</i> <i>String correcto,</i> <i>TpoDate fechaLimite,</i> <i>TpoPK expFase,</i> <i>String observaciones,</i> <i>TpoPK idDocExp,</i> <i>String numDoc,</i> <i>TpoPK idUniOrg,</i> <i>TpoPK idEstElab,</i> <i>String numDocOrig) throws TrException</i></p>	<i>Deprecado</i>
--	------------------

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido. Si el identificador es 'null' se generará una excepción.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>

<i>TpoDate</i> fecha	<i>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".</i>
<i>String</i> presentado	Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado
<i>String</i> correcto	Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto
<i>TpoDate</i> fechaLimite	Fecha límite de presentación o de corrección del documento.
<i>TpoPK</i> expFase	Identificador de la instancia de la fase
<i>String</i> observaciones	Observaciones para anotar a la hora de incorporar el documento.
<i>TpoPK</i> idDocExp	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente.</i>
<i>TpoPK</i> idUniOrg	<i>Entrada/Salida. Identificador del organismo de captura</i>
<i>TpoPK</i> idEstElab	<i>Identificador de estado de elaboración del documento ENI.</i>
<i>String</i> numDocOrig	<i>Número de documento origen en caso de que el documento ENI sea una copia con cambio de formato o una copia parcial.</i>

Salida

TrMensajeCondicionAccion [] Array de objetos *TrMensajeCondicionAccion*.

<p>TrMensajeCondicionAccion[] incorporarDocumento(<i>TpoPK</i> idExpte, <i>TpoPK</i> idDocPer, <i>TpoPK</i> idDefProc, <i>TpoDate</i> fecha, <i>String</i> presentado, <i>String</i> correcto, <i>TpoDate</i> fechaLimite, <i>TpoPK</i> expFase, <i>String</i> observaciones, <i>TpoPK</i> idDocExpte, <i>TpoPK</i> idEstElab, <i>String</i> identificadorENIOrg, <i>String</i> nombreFichero) throws <i>TrException</i></p>	<p><i>Deprecado</i></p>
--	-------------------------

Entradas

<i>TpoPK</i> idExpediente	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK</i> idDocPer	<i>Identificador del documento permitido. Si el identificador es 'null' se generará una excepción.</i>
<i>TpoPK</i> idDefProc	<i>Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>

<i>TpoDate</i> fecha	<i>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".</i>
<i>String</i> presentado	Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no. → " S " – El documento se ha presentado → " N " – El documento no se ha presentado
<i>String</i> correcto	Indica si el documento que se ha presentado es correcto o no. → " S " – El documento es correcto → " N " – El documento no es correcto
<i>TpoDate</i> fechaLimite	Fecha límite de presentación o de corrección del documento.
<i>TpoPK</i> expFase	Identificador de la instancia de la fase
<i>String</i> observaciones	Observaciones para anotar a la hora de incorporar el documento.
<i>TpoPK</i> idDocExp	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente.</i>
<i>TpoPK</i> idEstElab	<i>Entrada/Salida.</i> Identificador de estado de elaboración del documento ENI.
<i>String</i> identificadorENIOrg	<i>Número de documento origen en caso de que el documento ENI sea una copia con cambio de formato o una copia parcial.</i>
<i>String</i> nombreFichero	<i>Nombre con el que se almacena el fichero.</i>

Salida

<i>TrMensajeCondicionAccion</i> []	Array de objetos <i>TrMensajeCondicionAccion</i> .
-------------------------------------	--

2.4.1.4.1.53 Incorporar párrafos

```
void incorporarParrafosDocumento(
    TpoPK idDocExp,
    TrParrrafo [ ] datosParrrafo) throws TrException
```

Permite añadir nuevos párrafos al documento del expediente que se pasa como parámetro. Se permite indicar si el párrafo es editable o no.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento asociado al expediente. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TrParrafo[] datosParrafo</i>	<i>Conjunto de párrafos a añadir al documento.</i>

2.4.1.4.1.54 Insertar los datos de registro de documentos

```
void insertarDatosRegistroDocumentos (
    List<TpoPK> docsExp,
    TrDatosRegistroDocumento datosRegistro ) throws TrException
```

Permite añadir datos de registro para uno o varios documentos que se indica como parámetro (clave primaria de los documentos del expediente).

Entradas

<i>List<TpoPK> docsExp</i>	<i>Identificador de los documentos del expediente Si los identificadores no son válidos, se generará una excepción.</i>
<i>TrDatosRegistroDocument o datosRegistro</i>	<i>Datos de registro. Si el código de registros es 'null' se generará una excepción.</i>

2.4.1.4.1.55 Incorporar documento externo

```
TrMensajeCondicionAccion[] incorporarDocumentoExterno(
    TpoPK idExpediente,
    TpoPK idDocPer,
    TpoPK idDefProc,
    TpoDate fecha,
    String presentado,
    String correcto,
    TpoDate fechaLimite,
    TpoPK expFase,
    String observaciones,
    TpoPK idDocExpte
    TpoPK csv,
    String motivo,
    String nombreDoc) throws TrException
```

Permite al usuario incorporar un documento externo dado para un expediente en un procedimiento también suministrado, en caso de que el expediente bloqueado se podrá incorporar el documento siempre y cuando el atributo *comprobarBloqueo* sea false . Sólo se persistirán los metadatos del documento permaneciendo el contenido del mismo custodiado en el repositorio externo. El método devuelve un conjunto de mensajes/avisos derivados de las condiciones establecidas para la incorporación del documento.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido. Si el identificador es 'null' se generará una excepción.</i>

<i>TpoPK</i> idDefProc	Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoDate</i> fecha	Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".
<i>String</i> presentado	Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado
<i>String</i> correcto	Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto
<i>TpoDate</i> fechaLimite	Fecha límite de presentación o de corrección del documento.
<i>TpoPK</i> expFase	Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.
<i>String</i> observaciones	Observaciones para anotar a la hora de incorporar el documento.
<i>TpoPK</i> idDocExpte	Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.
<i>String</i> csv	Csv del documento original.
<i>String</i> motivo	Motivo para realizar la consulta al repositorio externo.
<i>String</i> nombreDoc	Nombre del documento.

Salida

TrMensajeCondicionAccion [] Array de objetos *TrMensajeCondicionAccion*.

2.4.1.4.1.56(w) Incorporar documento no definido

```

void incorporarDocumentoNoDefinido(
    TpoPK idExpediente,
    TpoPK idTipDoc,
    TpoPK idFase,
    TpoDate fecha,
    String presentado,
    String correcto,
    TpoDate fechaLimite,
    TpoPK expFase,
    String observaciones,
    TpoPK idDocExp,
    TpoPK idTipEni,
    TpoPK idEstElab,
    String numDocOrig) throws TrException

```

Permite incorporar un documento dado para un expediente no recogido en la definición del procedimiento.

Si se ha definido un componente *w@rdA* en el sistema establecido en el *TrAPIUI* también se guardará el documento incorporado en *w@rdA*.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idTipDoc</i>	<i>Identificador del tipo de documento que se quiere incorporar. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idFase</i>	<i>Identificador de la fase del expediente a la que se quiere asociar el documento incorporado. Debe ser una de las fases en las que se encuentre el expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoDate fecha</i>	<i>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado". Si no se indica fecha, por defecto se tomará la del sistema.</i>
<i>String presentado</i>	<i>Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado</i>
<i>String correcto</i>	<i>Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite de presentación o de corrección del documento.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String observaciones</i>	<i>Observaciones para anotar a la hora de incorporar el documento.</i>
<i>TpoPK idDocExpte</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.</i>
<i>TpoPK idTipEni</i>	<i>Identificador de tipo de expediente ENI</i>
<i>TpoPk idEstElab</i>	<i>Identificador del estado de elaboración del documento</i>
<i>String numDocOrig</i>	<i>Csv del documento original</i>

void incorporarDocumentoNoDefinido(

*TpoPK idExpediente,
TpoPK idTipDoc,
TpoPK idFase,
TpoDate fecha,
String presentado,
String correcto,
TpoDate fechaLimite,
String observaciones,
TpoPK idDocExp) throws TrException*

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idTipDoc</i>	<i>Identificador del tipo de documento que se quiere incorporar. Si el identificador es 'null', o no válido, se generará una excepción.</i>

<i>TpoPK idFase</i>	<p>Identificador de la fase del expediente a la que se quiere asociar el documento incorporado. Debe ser una de las fases en las que se encuentre el expediente. Si el identificador es 'null', o no válido, se generará una excepción.</p>
<i>TpoDate fecha</i>	<p>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado". Si no se indica fecha, por defecto se tomará la del sistema.</p>
<i>String presentado</i>	<p>Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado</p>
<i>String correcto</i>	<p>Indica si el documento que se ha presentado es correcto o no. → "S" – El documento es correcto → "N" – El documento no es correcto</p>
<i>TpoDate fechaLimite</i>	<p>Fecha límite de presentación o de corrección del documento.</p>
<i>String observaciones</i>	<p>Observaciones para anotar a la hora de incorporar el documento.</p>
<i>TpoPK idDocExpte</i>	<p>Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.</p>

void incorporarDocumentoNoDefinido(

TpoPK idExpediente,
 TpoPK idTipDoc,
 TpoPK idFase,
 TpoDate fecha,
 String presentado,
 String correcto,
 TpoDate fechaLimite,
 TpoPK expFase,
 String observaciones,
 TpoPK idDocExp) throws TrException

Entradas

<i>TpoPK idExpediente</i>	<p>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</p>
<i>TpoPK idTipDoc</i>	<p>Identificador del tipo de documento que se quiere incorporar. Si el identificador es 'null', o no válido, se generará una excepción.</p>
<i>TpoPK idFase</i>	<p>Identificador de la fase del expediente a la que se quiere asociar el documento incorporado. Debe ser una de las fases en las que se encuentre el expediente. Si el identificador es 'null', o no válido, se generará una excepción.</p>
<i>TpoDate fecha</i>	<p>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado". Si no se indica fecha, por defecto se tomará la del sistema.</p>
<i>String presentado</i>	<p>Indica si el documento se ha presentado o no. → "S" – El documento se ha presentado → "N" – El documento no se ha presentado</p>

<i>String</i> correcto	Indica si el documento que se ha presentado es correcto o no. → “ S ” – El documento es correcto → “ N ” – El documento no es correcto
<i>TpoDate</i> fechaLimite	Fecha límite de presentación o de corrección del documento.
<i>TpoPK expFase</i>	Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.
<i>String</i> observaciones	Observaciones para anotar a la hora de incorporar el documento.
<i>TpoPK idDocExpte</i>	<i>Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.</i>

<p>void incorporarDocumentoNoDefinido(TpoPK idExpediente, TpoPK idTipDoc, TpoPK idFase, TpoDate fecha, String presentado, String correcto, TpoDate fechaLimite, TpoPK expFase, String observaciones, TpoPK idDocExp, TpoPK idTipEni, TpoPK idEstElab, String numDocOrig) throws TrException</p>	<i>Deprecado</i>
--	------------------

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idTipDoc</i>	<i>Identificador del tipo de documento que se quiere incorporar. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idFase</i>	<i>Identificador de la fase del expediente a la que se quiere asociar el documento incorporado. Debe ser una de las fases en las que se encuentre el expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoDate fecha</i>	<i>Fecha de incorporación del documento. Es la fecha en la cual el documento queda anotado para el expediente como "incorporado".</i>
<i>String</i> presentado	<i>Si no se indica fecha, por defecto se tomará la del sistema. Indica si el documento se ha presentado o no. → “S” – El documento se ha presentado → “N” – El documento no se ha presentado</i>
<i>String</i> correcto	<i>Indica si el documento que se ha presentado es correcto o no. → “S” – El documento es correcto → “N” – El documento no es correcto</i>
<i>TpoDate</i> fechaLimite	<i>Fecha límite de presentación o de corrección del documento.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>

String observaciones	Observaciones para anotar a la hora de incorporar el documento.
TpoPK idDocExpte	Identificador único que TREW@ le da al documento incorporado para ese expediente. Entrada/Salida.
TpoPK idTipEni	Identificador de tipo de expediente ENI
TpoPk idEstElab	Identificador del estado de elaboración del documento
String numDocOrig	Csv del documento original

2.4.1.4.1.57(w) Modificar los datos de un documento del expediente

```
void modificarDatosDocumento(
    TrDocumentoExpediente datosDocExp) throws TrException
```

Permite modificar los datos principales asociados a un documento generado o incorporado siempre que se tengan permisos para hacerlo (no esté reservada por otro usuario la fase dónde el documento esté permitido, etc.).

Si se define un componente w@rdA en el sistema establecido en el TrAPIUI y el documento está guardado en el mismo w@rdA se actualizan sus datos.

Entradas

TrDocumentoExpediente
datosDocExpte

Objeto TrDocumentoExpediente con la información asociada a un documento generado y/o incorporado a un expediente.

Se precisa rellenar los siguientes atributos de TrDocumentoExpediente:
 → REFDOCEXP – Identificador del documento a modificar.
 → FECHA / PRESENTADO / CORRECTO / FECHA_LIMITE / OBSERVACIONES / C_HASH / FECHACADU / REUTILIZABLE / FIRMADIG

2.4.1.4.1.58 Modificar los datos de un documento del expediente

```
void modificarDatosDocumentoExterno( TpoPK idDoc,
    String refDocExterno,
    String refAnexoDocExterno,
    TpoPK idComponente ) throws TrException
```

Permite modificar las referencia del documento en un gestor documental externo (w@rdA). Modifica el valor de todos los parámetros, es decir si se le pasa *null* se actualizará el valor de la columna a *null*.

Entradas

TpoPK idDoc

Identificador del documento del expediente

String refDocExterno

Identificador en el gestor documental externo del documento del expediente en Trew@.

String refAnexoDocExterno

Identificador en el gestor documental externo del anexo del documento del expediente en Trew@.

	Junta de Andalucía Agencia Digital de Andalucía	Trew@ Guía de referencia de JTrApi
---	---	---------------------------------------

TpoPK idComponente

Identificador del componente gestor documental.

2.4.1.4.1.59 Incluir documento

```
void incluirDocumento(
    TpoPK idDocExpteOrigen,
    TpoPK idDocExpteDestino,
    long ordenParrafoInicio) throws TrException
```

Incluye los párrafos perteneciente al cuerpo del documento origen, en un documento destino a partir de un nº de orden inicial desplazando los párrafos del documento destino.

Entradas

<i>TpoPK idDocExpteOrigen</i>	<i>Identificador del documento origen. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocExpteDestino</i>	<i>Identificador del documento destino. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>long ordenParrafoInicio</i>	<i>Orden, en el documento destino, a partir del cual se va a insertar el documento origen.</i>

2.4.1.4.1.60 Obtener documento ENI mediante CSV

```
DocumentoEni incluirDocumento(
    String csv,
    TpoString motivo) throws TrException
```

Obtiene el documento ENI por referencia CSV.

Entradas

<i>TpoPK idDocExpteOrigen</i>	<i>Identificador del documento origen. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocExpteDestino</i>	<i>Identificador del documento destino. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>long ordenParrafoInicio</i>	<i>Orden, en el documento destino, a partir del cual se va a insertar el documento origen.</i>

Salidas *DocumentoEni*

Obtiene el documento ENI

2.4.1.4.1.61(w) Modificar interesado en un documento de un expediente

```

void modificarInteresadoDocumento(
    TpoPK idExpediente,
    TpoPK idInteresado,
    TpoPK idRazonInt,
    TpoPK idRazonIntDocu,
    TpoPK idDocExp,
    TpoPK idRazonIntDocuNueva,
    String observaciones) throws TrException

```

Permite modificar la razón de interés del documento, siempre que no tenga notificaciones, y las observaciones del interesado en el documento. Se comprueba que existan todos los parámetros y además que el expediente no esté archivado.

Si se define un componente w@rdA en el sistema establecido en el TrAPIUI y el documento del que se modifica el interesado está guardado en el mismo w@rdA se actualizan sus datos.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idInteresado</i>	<i>Identificador del interesado Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idRazonInt</i>	<i>Identificador de la razón de interés del interesado del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idRazonIntDocu</i>	<i>Identificador de la razón de interés del interesado en el documento. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idDocExp</i>	<i>Identificador del documento del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idRazonIntDocuNueva</i>	<i>Identificador de la nueva razón de interés del interesado en el documento. Si el identificador es 'null' no se actualiza.</i>
<i>String observaciones</i>	<i>Observaciones para el nuevo interesado en el documento</i>

```

void modificarInteresadoDocumento(
    TpoPK idExpediente,
    TpoPK idInteresado,
    TpoPK idRazonInt,
    TrDatosContacto datosContacto,
    TpoPK idRazonIntNueva,
    String observaciones) throws TrException

```

Deprecado

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idInteresado</i>	<i>Identificador del interesado Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idRazonInt</i>	<i>Identificador de la razón de interés del interesado del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TrDatosContacto datosContacto TpoPK idRazonIntNueva</i>	<i>Identificador de la razón de interés Identificador de la nueva razón de interés del interesado en el documento. Si el identificador es 'null' no se actualiza.</i>
<i>String observaciones</i>	<i>Observaciones para el nuevo interesado en el documento</i>

2.4.1.4.1.62 Modificar envío notificación

<p>void modificarNotificacionEnvio(TrNotificacionEnvio notificacionEnvio) throws TrException</p>
--

Permite modificar los datos de un envío de notificación existente de un interesado en un documento del expediente.

Entradas

<i>TrNotificacionEnvio notificacionEnvio</i>	<i>Envío de notificación a modificar.</i>
--	---

2.4.1.4.1.63 Modificar observaciones en documento expediente

<p>void modificarObservacionesDocumentoExpediente(TpoPK idDocExp, String observaciones) throws TrException</p>
--

Modifica las observaciones asociadas a un documento expediente.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador de documento expediente</i>
<i>String observaciones</i>	<i>Observaciones del documento</i>

2.4.1.4.1.64 Modificar orden párrafo

```
void modificarOrdenParrafoDocumento( TpoPK idParrafoDocExp,  
long orden) throws TrException
```

Permite modificar sólo el orden de un párrafo de un documento. No se tiene en cuenta si es "editable".

Entradas

<i>TpoPK idParrafoDocExp</i>	Identificador del párrafo del documento, Si el identificador es 'null' o no válido se generará una excepción.
<i>long orden</i>	Orden para el párrafo.

2.4.1.4.1.65 Modificar párrafos

```
void modificarParrafoDocumento(  
TrParrafo datosParrafo ) throws TrException
```

Permite modificar los datos de un párrafo de un documento generado siempre que sea editable. El método no modifica la condición de "Editable".

2.4.1.4.1.66 Acciones asociadas a un documento

```
TrAccionDocumento[ ] obtenerAccionesDocumento(  
TpoPK idDefProc,  
TpoPK idDocPer,  
ClausulaWhere where,  
ClausulaOrderBy orderBy ) throws TrException
```

Permite al usuario obtener los datos del conjunto de acciones (válidas) asociadas a un determinado documento en una fase y un procedimiento dados.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento</i> <i>Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDocPer</i>	<i>Identificador del documento</i> <i>Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> <i>Consultar campos posibles para TrAccionDocumento.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> <i>Consultar campos posibles para TrAccionDocumento.</i>

Salida

<i>TrAccionDocumento []</i>	<i>Array de objetos TrAccionDocumento.</i>
------------------------------	--

2.4.1.4.1.67 Obtener certificados

TrTipoCertificado[] obtenerCertificados(TpoPK idTipoDoc,
TpoPK idSistema,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener certificados asociados a tipos de documentos. Si se pasa el parámetro *idTipoDoc* devuelve los asociados a ese tipo de documento, si se pasa el parámetro *idSistema* devuelve los certificados asociados a ese sistema, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK idTipoDoc</i>	Identificador del tipo de documento asociado a los tipos de certificado. Si se pasa <i>null</i> se obtienen todos.
<i>TpoPK idSistema</i>	Identificador del sistema del que queremos obtener los tipos de certificado. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrTipoCertificado
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrTipoCertificado

Salida

<i>TrTipoCertificado[]</i>	Array de objetos TrTipoCertificado
-----------------------------	------------------------------------

2.4.1.4.1.68 Condiciones asociadas a un documento

TrCondicionDocumento[] obtenerCondicionesDocumento(
TpoPK idDefProc,
TpoPK idDocPer,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite al usuario obtener los datos del conjunto de condiciones (válidas) asociadas a un determinado documento en una fase y un procedimiento dados.

Entradas

<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idDocPer</i>	Identificador del documento Si el identificador es 'null', o no válido, se generará una excepción.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrCondicionDocumento.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrCondicionDocumento.

Salida

TrCondicionDocumento [] *Array de objetos TrCondicionDocumento.*

2.4.1.4.1.69 Obtener consultas de certificados de interesados en un documento

TrConsultaCertificado[] obtenerConsultaCertificado(
 TpoPK idConsulta,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener un conjunto de consultas de certificados para los que el interesado en un documento ha dado su consentimiento y una vez consultado se guardan los datos de dicha consulta en Trew@.

Entradas

<i>TpoPK idConsulta</i>	<i>Identificador de la consulta</i> Si no se indica se devuelve todas
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para <i>TrConsultaCertificado</i> .
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para <i>TrConsultaCertificado</i> .

Salida

TrConsultaCertificado [] *Array de objetos TrConsultaCertificado*

2.4.1.4.1.70 Datos de registro de documentos

TrDatosRegistroDocumento[] obtenerDatosRegistrosDocumento (
 TpoPK idDocExpte
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Devuelve los datos de registro del documento indicado como parámetro (clave primaria del documento del expediente).

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento del expediente</i> <i>Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para <i>TrDatosRegistroDocumento</i> .
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para <i>TrDatosRegistroDocumento</i> .

2.4.1.4.1.71 Obtener DocumentoEni externo

DocumentoEni ObtenerDocumentoEniExterno(
 String csv,
 String motivo) throws TrException

Permite obtener un documento custodiado en un repositorio externo indicando el csv y el motivo como parámetros.

Entradas

String csv Csv del documento custodiado en repositorio externo.
String motivo Motivo para realizar la consulta al repositorio externo.

Salida

DocumentoEni Objeto DocumentoEni

2.4.1.4.1.72 Obtiene los documentos expedientes del hijo

Map<TpoPK, TrDocumentoExpediente[]> obtenerDocumentosExpedientesHijo(
 TpoPK idExpediente,
 boolean soloUsuario,
 ClausulaWhere where,
 ClausulaOrderBy orderBy,
 ClausulaExclusion exclusiones) **throws TrException**

Obtiene los documentos expedientes del hijo.

Entradas

TpoPK idExpediente Identificador del expediente padre
 Boolean soloUsuario Si es solo usuario
 ClausulaWhere where Cláusula where
 ClausulaOrderBy orderBy Cláusula de orden
 ClausulaExclusion exclusiones Cláusula de exclusión

2.4.1.4.1.73 Obtener EniDoc externo

EniDoc obtenerEniDocExterno(
 String csv,
 String motivo) **throws TrException**

Permite obtener un objeto EniDoc del documento custodiado en un repositorio externo indicando el csv y el motivo como parámetros.

Entradas

String csv Csv del documento custodiado en repositorio externo.
String motivo Motivo para realizar la consulta al repositorio externo.

Salida

EniDoc Objeto EniDoc

2.4.1.4.1.74 Obtener envíos notificaciones

TrNotificacionEnvio [] obtenerNotificacionesEnvio(
 TpoPK idNotEnv,
 TpoPK idRem,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener los envíos de notificaciones de los interesados en los documentos de un expediente. Si no indicamos ningún parámetro se obtendrán todos los envíos de notificación existentes en el sistema.

Entradas

<i>TpoPK idNotEnv</i>	<i>Identificador de notificación envío</i> <i>Si el identificador es 'null' se obtendrán todos los envíos de notificación existentes en el sistema.</i>
<i>TpoPK idRem</i>	<i>Identificador de remesa.</i> <i>Opcional.</i>
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrNotificacionEnvio.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrNotificacionEnvio.

Salida

TrNotificacionEnvio [] *Array de objetos TrNotificacionEnvio*

2.4.1.4.1.75 Recuperar párrafos

TrParrafo[] obtenerParrafosDocumento(
 TpoPK idDocExpte,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite recuperar los párrafos de un documento generado. Por defecto ordena por el campo "orden" de los párrafos siempre y cuando no se le pase ninguna ClausulaOrderBy.

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento asociado al expediente.</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrParrafo.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrParrafo. Si no se indica ordenación, la misma por defecto es por el orden de los párrafos.

Salida

TrParrafo[] *Array de objetos TrParrafo con la información de tipos de expediente.*

2.4.1.4.1.76 Recuperar párrafos de tipo documento

TrParrafoTipoDocumento[] obtenerParrafosTipoDocumento(
 TpoPK idTipoDocumento
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite recuperar todos los párrafos del tipo de documento indicado.

Entradas

<i>TpoPK idTipoDocumento</i>	<i>Identificador del tipo de documento</i> <i>Si el identificador es 'null' se devolverán todos.</i>
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrParrafoTipoDocumento.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrParrafoTipoDocumento.

Salida

<i>TrParrafoTipoDocumento []</i>	Array de objetos TrParrafoTipoDocumento con la información de tipos de expediente.
-----------------------------------	--

2.4.1.4.1.77 Evaluar párrafo

String sustituirVariablesParrafoDocumento(
 TpoPK idSistema,
 TpoPK idParrDocExp) throws TrException

Devuelve el párrafo con el valor de las variables según los parámetros del documento que le pasamos.

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema.</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idParrDocExp</i>	<i>Identificador del párrafo del documento.</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>

Salida

<i>String</i>	Párrafo con las variables reemplazadas.
---------------	---

2.4.1.4.1.78 Comprobar si el tipo de documento es editable

boolean comprobarEditable(
 String idTipoDocExp) throws TrException

Comprueba si el tipo de documento es editable por el tramitador una vez se haya generado y sustituido las variables.

Entradas

String idTipoDocExp Identificador del tipo de documento.

2.4.1.4.1.79 Consentimiento para la consulta de un certificado

<p>boolean consentimientoConsultaCertificado(TpoPK idExpediente TpoPK idInteresado, TpoPK idRazonInt, TpoPK idRazonIntDocu, TpoPK idDocExp) throws TrException</p>
--

Indica si se tiene consentimiento para realizar la consulta del certificado. El consentimiento puede ser indicado por el propio interesado o ser por Ley.

Entradas

<i>TpoPK idExpte</i>	<i>Identificador del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idInte</i>	<i>Identificador del interesado Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idRazonInt</i>	<i>Identificador de la razón de interés en el expediente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idRazonIntDocu</i>	<i>Identificador de la razón de interés en el documento Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idDocExp</i>	<i>Identificador del documento del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>

2.4.1.4.1.80(b) Eliminar interesado

<p>void eliminarInteresado(TpoPK idInteresado, boolean elimRelaInt) throws TrException</p>
--

Permite eliminar el interesado que se indique en el parámetro *idInteresado* así como sus relaciones, según se indique en el parámetro *elimRelaInt*.

Si el sistema establecido tiene conexión al bus y el interesado tiene *ciw@* (Código identificador *w@ndA*) se informará del borrado del interesado.

Entradas

<i>TpoPK idInteresado</i>	<i>Identificador del interesado Si el identificador es 'null', o no válido se generará una excepción</i>
<i>boolean elimRelaInt</i>	<i>Indica si se eliminan las relaciones del interesado. - true – Se eliminan las relaciones - false – No se eliminan las relaciones</i>

2.4.1.4.1.81 Eliminar envío notificación

void eliminarNotificacionEnvio(TrInteresadoDocumento intDoc,
TpoPK idNotEnv) throws TrException

Permite eliminar una notificación asociada a un interesado en documento. Se comprueba que el interesado en documento existe y que el expediente no esté archivado.

Si el envío de notificación no está asociado a ningún otro interesado en documento será eliminado también.

Entradas

<i>TrInteresadoDocumento</i> intDoc	<i>Interesado en documento del que queremos eliminar la notificación.</i>
<i>TpoPK idNotEnv</i>	<i>Identificador de envío de notificación asociado al interesado en documento.</i>

2.4.1.4.1.82(w) Eliminar interesado en un documento de un expediente

void eliminarInteresadoDocumento(
TpoPK idExpediente,
TpoPK idInteresado,
TpoPK idRazonInt,
TpoPK idRazonIntDocu
TpoPK idDocExp,
boolean elimNotif) throws TrException

Permite eliminar los interesados en los documentos del expediente. Se comprueba que existan todos los parámetros, excepto el identificador de la razón de interés del documento, y además que el expediente no esté archivado.

Se da la posibilidad de eliminar las notificaciones asociadas al interesado en el documento. Para ello se usa el parámetro *elimNotif* con los siguientes valores:

- **true:** Indica que se eliminen las notificaciones, en caso de que hubiese.
- **false:** Indica que no se eliminen las notificaciones, generando una excepción en caso de que hubiese.

Si se define un componente *w@rdA* en el sistema establecido en el TrAPIUI y el documento del que se elimina el interesado está guardado en el mismo *w@rdA* se elimina también de éste.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idInteresado</i>	<i>Identificador del interesado Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idRazonInt</i>	<i>Identificador de la razón de interés del interesado del expediente Si el identificador es 'null', o no válido, se generará una excepción</i>

<i>TpoPK idRazonIntDocu</i>	<i>Identificador de la razón de interés del interesado en el documento.</i>
<i>TpoPK idDocExp</i>	<i>Opcional Identificador del documento del expediente Si el identificador es 'null' no se actualiza.</i>
<i>boolean elimNotif</i>	<i>Indica si se eliminan las notificaciones del interesado en el documento. true: Se eliminan las notificaciones false: No se eliminan las notificaciones, generando una excepción en caso de que tuviese.</i>

2.4.1.4.1.83 Eliminar párrafos

void eliminarParrafoDocumento(
TpoPK idParrafoDoc) throws TrException

Permite al usuario eliminar un párrafo de un documento siempre que tenga permisos.

Entradas

<i>TpoPK idParrafoDoc</i>	<i>Identificador del párrafo a eliminar. Si el identificador es 'null', o no válido, se generará una excepción</i>
---------------------------	--

2.4.1.4.1.84 Comprobar si es documento de repositorio externo

boolean esDocumentoRepositorioExterno(TpoPK idDocExpte) throws TrException

Permite comprobar si el documento pertenece a un documento existente en un repositorio externo.

Entradas

<i>TpoPK idDocExpte</i>	<i>Identificador del documento.</i>
-------------------------	-------------------------------------

2.4.1.4.1.85 Establecer configuración para la generación de pdf

boolean establecerConfiguracionGeneracionPdf(TpoPK idDocExp) throws TrException

Establece la configuración de la plantilla xml del tipo de documento del documento indicado mediante el parámetro idDocExp. Para ello hace uso de la constante del sistema URL_REPORT_JAVA en la que se define la ruta del servidor donde se encuentra la plantilla y recursos necesarios para la generación de documentos Java PDF.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento del expediente.</i>
-----------------------	--

2.4.1.4.1.86 Generar finalidad SCSP

String generarFinalidadSCSP(TpoPK idExpediente, TrDocumentoPermitido docPer) throws TrException

Permite obtener la finalidad para realizar las consultas para los tipos de certificados SCSP de los procedimientos.

Entradas

<i>TpoPK idExpediente</i>	Identificador del expediente para realizar la petición SCSP.
<i>TrDocumentoPermitido docPer</i>	Documento permitido al que pertenece el tipo de certificado.

Salida

<i>String</i>	Cadena con la petición SCSP
---------------	-----------------------------

2.4.1.4.1.87 Genera pdf resumen del expediente

byte[] generaPdfResumen(
String refExp,
boolean infoAuxPdfResumen) throws TrException

Resumen del expediente en formato PDF que incorpora toda la información relevante del expediente: los documentos vinculados a su índice, su firma y sus metadatos

Entradas

<i>String refExp</i>	Identificador del expediente.
<i>boolean infoAuxPdfResumen</i>	Indica si se quiere mostrar información auxiliar de los documentos pertenecientes al expediente.

Salida

<i>byte[]</i>	Contenido binario del pdf resumen.
---------------	------------------------------------

2.4.1.4.1.88 Insertar envío notificación

void insertarNotificacionEnvio(TrInteresadoDocumento[] intsDoc,
TrNotificacionEnvio notificacionEnv) throws TrException

Permite insertar un nuevo envío de notificación para un conjunto de interesados en documento del expediente. Se comprueba que existan los interesados en documento, que los documento sean notificables, que no se haya superado el número de envíos para cada uno de ellos y que el expediente no esté archivado.

Entradas

<i>TrInteresadoDocumento[] instDoc</i>	<i>Interesados en documentos a los cuales se les asociará el envío de notificación.</i>
<i>TrNotificacionEnvio notificacionEnv</i>	<i>Envío de notificación a insertar para los interesados en documentos.</i>

2.4.1.4.1.89 Insertar consulta de certificado de un interesado en un documento

TpoPK insertarConsultaCertificado(TrConsultaCertificado consultaCert) throws TrException

Permite insertar los datos de una nueva consulta de certificados para un interesado en un documento del expediente.

Entradas

<i>TrConsultaCertificado consultaCert</i>	Consulta del certificado. El REFCONSULTA se rellenará con el identificador que se asigna al registro insertado
---	--

Salidas

<i>TpoPK</i>	Identificador de la consulta. También se puede consultar en el mismo objeto que se pasa mediante el atributo REFCONSULTA.
--------------	---

2.4.1.4.1.90 Modificar consulta de certificado de un interesado en un documento

void modificarConsultaCertificado(TrConsultaCertificado consultaCert) throws TrException

Entradas

<i>TrConsultaCertificado consultaCert</i>	Consulta del certificado.
---	---------------------------

2.4.1.4.1.91 Modificar datos registro documentos

```

void modificarDatosRegistroDocumento(
    TpoPK idDocExp,
    String numRegistroActual,
    TrRegistroDocumento datosRegistro) throws TrException
    
```

Deprecado

Permite modificar los datos de registro para el documento que se indica como parámetro.

Entradas

TpoPK idDocExp	Identificador del documento que se quiere modificar
String numRegistroActual	Número registro del documento que se quiere modificar
TrRegistroDocumento datosRegistro	Datos del registro para modificar

```

void modificarDatosRegistroDocumento(
    TpoPK idDocExp,
    TrRegistroDocumento datosRegistro) throws TrException
    
```

Deprecado

Entradas

TpoPK idDocExp	Identificador del documento que se quiere modificar
TrRegistroDocumento datosRegistro	Datos del registro para modificar

2.4.1.4.1.92 Obtención de plantillas del tipo de documento

```

TrPlantillaTipoDocumento[ ] obtenerPlantillasTipoDocumento( TpoPK idTipoDoc
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException
    
```

Devuelve el conjunto de plantillas del tipo de documento indicado.

Entradas

<i>TpoPK idTipoDoc</i>	<i>Identificador del tipo de documento. Si el identificador es 'null' se devolverán todos.</i>
------------------------	--

ClausulaWhere where

Filtro a aplicar.

Consultar campos posibles para TrPlantillaTipoDocumento.

ClausulaOrderBy orderBy

Ordenación a aplicar.

Consultar campos posibles para TrPlantillaTipoDocumento.

Salida

TrPlantillaTipoDocumento [] Array de objetos TrPlantillaTipoDocumento.

2.4.1.4.1.93Tipos de documentos

TrTipoDocumento[] obtenerTiposDocumento(
 TpoPK idStma,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Devuelve el conjunto de datos sobre los tipos de documento definidos en el sistema, que se pasa como parámetro. Si no se indica sistema devuelve todos los tipos de documentos definidos en el tramitador.

Entradas

TpoPK idStma

*Identificador de la definición del procedimiento
Opcional*

ClausulaWhere where

Filtro a aplicar.

Consultar campos posibles para TrTipoDocumento.

ClausulaOrderBy orderBy

Ordenación a aplicar.

Consultar campos posibles para TrTipoDocumento.

Salida

TrTipoDocumento []

Array de objetos TrTipoDocumento.

2.4.1.4.1.94Recuperar documento sellado

byte[] recuperarDocSellado(
 TpoPK idDocExp,
 byte[] doc) throws TrException

Recupera el documento expediente sellado.

Entradas

TpoPK idDocExp

Identificador del documento expediente

byte[] doc

Documento

Salida

byte []

Array de byte .

2.4.1.4.1.95Insertar sello registro salida nuevo

byte[] insertarSelloRegistroSalidaNuevo(
 TrDatosRegistroDocumento iregistroDoc,
 byte[] docuPortafirmas) throws SelloException, IOException

Establece el nuevo sello de registro salida.

Entradas

TrDatosRegistroDocumento iregistroDoc *Clase que representa la información de registro para un documento*

byte[] docuPortafirmas *Documento port@firmas*

Salida

byte [] *Array de byte .*

2.4.1.4.1.96 Insertar sello registro entrada nuevo

```
byte[] insertarSelloRegistroEntradaNuevo(
    TrDatosRegistroDocumento iregistroDoc,
    byte[] docuPortafirmas) throws SelloException, IOException
```

Establece el nuevo sello de registro entrada.

Entradas

TrDatosRegistroDocumento iregistroDoc *Clase que representa la información de registro para un documento*

byte[] docuPortafirmas *Documento port@firmas*

Salida

byte [] *Array de byte .*

2.4.1.4.1.97 Editar nombre de un documento incorporado.

```
void editarNombreDocumento(
    TpoPK idDocExpte,
    String tituloDocExp) throws TrException
```

El método permite modificar el nombre de un documento ya incorporado. Esta modificación sólo será posible si el documento no se encuentra firmado, si el documento no está registrado en @ries y si el documento no está borrado. Con respecto al nuevo nombre, no debe superar los 250 caracteres y controla si el nuevo nombre contiene caracteres especiales o no, si los contiene los sustituye por una "X".

Entradas

TpoPK idDocExpte *Id del documento al que se le va a modificar el nombre.*

String tituloDocExp *Documento port@firmas*

2.4.1.4.1.98 Validar el Tipo Identificador de un Interesado

```
void validacionTipIdentificador(
    TrInteresado interesado) throws TrException
```



```
void validacionTipoIdentificador(  
    TrInteresadoSimple interesadoSimple) throws TrException
```

```
void validacionTipoIdentificador(  
    TrInteresado interesado,  
    TrInteresadoSimple interesadoSimple) throws TrException
```

Los dos primeros métodos sirven para llamar al tercer método que es el que realiza la validación. El tercer método asegura que el tipo identificador del Interesado o InteresadoSimple que tiene asignado sea el adecuado en función del número identificador configurado, siguiendo la siguiente lógica:

1. Si el número de identificador es de tipo NIF o NIE, el tipo identificador será del mismo tipo que el número de identificador.
2. Si el número de identificador no es de tipo NIF o NIE, el tipo identificador se mantendrá siempre y cuando no sea NIF o NIE.
 1. Si fuera NIF o NIE lanzará el error: "El identificador introducido no corresponde al tipo identificador indicado."
3. Si el número de identificador no es de tipo NIF o NIE, el tipo identificador no está configurado y existe en el modelo de datos el tipo identificador "OTRO", se asignará este al interesado recibido.
 1. Si no existe en el modelo de datos el tipo identificador "Otro", lanzará el error: "El identificador no es de tipo NIF o NIE y no se le puede asignar OTRO tipo de identificador."

Entradas

TrInteresado interesado *Interesado a validar*

TrInteresadoSimple
interesadoSimple *InteresadoSimple a validar*

2.4.1.4.2 APIs sobre firmas de documentos

2.4.1.4.2.1 Firmar documento interesado

```
void firmarDocumentoInteresado(  
    TpoPK idDocExpte,  
    TpoPK idInteresado  
    TpoDate fecha,  
    String editable,  
    String txtPie,  
    String txtFdo,  
    Int orden,  
    String codTransaccion,  
    Byte[] pkcs7,
```



String c_hash,
String tipoFirmaEni) throws TrException

Permite firmar un documento que se pasa como parámetro.

Se podrá firmar siempre que el interesado este asociado al documento en el expediente.

Entradas

<i>TpoPK idDocExpte</i>	Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoPK idInteresado</i>	Identificador del interesado. Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoDate fecha</i>	Fecha de la firma. Si no se indica nada, por defecto toma la del sistema.
<i>String editable</i>	Indica si el pie de firma es editable o no. → " S " – Sí es editable → " N " – No es editable
<i>String txtPie</i>	Texto de pie de firma.
<i>String txtFdo</i>	Texto firmado.
<i>String codTransaccion</i>	Código de la transacción de firma
<i>Byte[] pkcs7</i>	Recibo de la firma digital
<i>String c_hash</i>	Código hash de la firma digital
<i>String tipoFirmaEni</i>	Indica el tipo de firma ENI. → TF01- CSV → TF02 - XAdES internally detached signature. → TF03 - XAdES enveloped signature. → TF04 - CAdES detached/explicit signature. → TF05 - CAdES attached/implicit signature. → TF06 - PAdES.

void firmarDocumentoInteresado(

TpoPK idDocExpte,
TpoPK idInteresado
TpoDate fecha,
String editable,
String txtPie,
String txtFdo,
Int orden,
String codTransaccion,
Byte[] pkcs7,
String c_hash) throws TrException

Deprecado

Entradas

<i>TpoPK idDocExpte</i>	Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoPK idInteresado</i>	Identificador del interesado. Si el identificador es 'null', o no válido, se generará una excepción

<i>TpoDate fecha</i>	Fecha de la firma. Si no se indica nada, por defecto toma la del sistema.
<i>String editable</i>	Indica si el pie de firma es editable o no. → “ S ” – Sí es editable → “ N ” – No es editable
<i>String txtPie</i>	Texto de pie de firma.
<i>String txtFdo</i>	Texto firmado.
<i>String codTransaccion</i>	Código de la transacción de firma
<i>Byte[] pkcs7</i>	Recibo de la firma digital
<i>String c_hash</i>	Código hash de la firma digital

2.4.1.4.2 Firmar documento

```

void firmarDocumento(
    TpoPK idDocExp,
    TpoDate fecha,
    String idPuestoTrab,
    TpoPK idUniOrg,
    String usuarioPTUO,
    String editable,
    String txtPie,
    String txtFdo
    String tipoFirmaEni) throws TrException

```

Permite firmar un documento que se pasa como parámetro.

Se podrá firmar siempre que se tengan permisos para ello en el tramitador. Si el filtro de firmantes está desactivado no se comprobará el número de firmas con el número de firmantes.

Entradas

<i>TpoPK idDocExp</i>	Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoDate fecha</i>	Fecha de la firma. Si no se indica nada, por defecto toma la del sistema.
<i>String idPuestoTrab</i>	Identificador del puesto de trabajo del firmante.
<i>TpoPK idUniOrg</i>	Identificador de la unidad orgánica del firmante, en caso de que el usuario que llama a la función no lo sea.
<i>String usuarioPTUO</i>	Usuario que ocupa el puesto de trabajo.
<i>String editable</i>	Indica si el pie de firma es editable o no. → “ S ” – Sí es editable → “ N ” – No es editable
<i>String txtPie</i>	Texto de pie de firma.
<i>String txtFdo</i>	Texto firmado.

String tipoFirmaEni

Indica el tipo de firma ENI.
 → TF01- CSV
 → TF02 - XAdES internally detached signature.
 → TF03 - XAdES enveloped signature.
 → TF04 - CADES detached/explicit signature.
 → TF05 - CADES attached/implicit signature.
 → TF06 - PAdES.

Si el usuario que llama a la función no se corresponde con el firmante, entonces se estará hablando de una "firma manuscrita" y será necesario indicar el puesto de trabajo y la unidad orgánica del firmante.

Ya que varios usuarios pueden ocupar el mismo puesto, indicando el usuario que ocupa el puesto, se dice qué usuario es realmente el que firma.

<p>void firmarDocumento(TpoPK idDocExp, TpoDate fecha, String idPuestoTrab, TpoPK idUniOrg, String usuarioPTUO, String editable, String txtPie, String txtFdo) throws TrException</p>	<i>Deprecado</i>
---	------------------

Entradas

<i>TpoPK idDocExp</i>	Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoDate fecha</i>	Fecha de la firma. Si no se indica nada, por defecto toma la del sistema.
<i>String idPuestoTrab</i>	Identificador del puesto de trabajo del firmante.
<i>TpoPK idUniOrg</i>	Identificador de la unidad orgánica del firmante, en caso de que el usuario que llama a la función no lo sea.
<i>String usuarioPTUO</i>	Usuario que ocupa el puesto de trabajo.
<i>String editable</i>	Indica si el pie de firma es editable o no. → " S " – Sí es editable → " N " – No es editable
<i>String txtPie</i>	Texto de pie de firma.
<i>String txtFdo</i>	Texto firmado.

2.4.1.4.2.3 Firmar documentos

<p>void firmarDocumentos(List<TrFirmaDocumentoExpediente> lineasDeFirma) throws TrException</p>
--

Permite firmar un conjunto de documentos a partir de líneas de firma pasadas por parámetros.

2.4.1.4.2.4 Delegar sustituir firma

```
void delegarSustituirFirmante(
    String idPtoTrab,
    TpoPK idUniOrg,
    String idPtoTrabDelegSust,
    TpoPK idUniOrgDelegSust,
    TpoPK idTeDi,
    TpoDate fecha,
    TpoDate fechaFin,
    String tipo,
    TpoPK idTipoDoc) throws TrException
```

Permite establecer delegaciones/sustituciones de firma en el sistema.

Entradas

<i>String idPtoTrab</i>	<i>Identificador del puesto de trabajo del firmante principal.</i>
<i>TpoPK idUniOrg</i>	<i>Identificador de la unidad orgánica del firmante principal</i>
<i>String idPtoTrabDelegSust</i>	<i>Identificador del puesto de trabajo del delegado/sustituto</i>
<i>TpoPK idUniOrgDelegSust</i>	<i>Identificador de la Unidad orgánica del delegado/sustituto</i>
<i>TpoPK idTeDi</i>	<i>Identificador de textos disposiciones</i>
<i>TpoDate fecha</i>	<i>Fecha en que comienza la sustitución. Si no se indica nada, por defecto toma la del sistema.</i>
<i>TpoDate fechaFin</i>	<i>Fecha en la que termina la sustitución.</i>
<i>String tipo</i>	<i>Tipo → "D" – Delegación (valor por defecto) → "S" – Sustitución</i>
<i>TpoPK idTipoDoc</i>	<i>Identificador del tipo de documento en el que se delega (para el caso de delegaciones).</i>

2.4.1.4.2.5 Eliminar firma digital

```
void eliminarFirmaDigital(
    TpoPK idDocExp,
    String usuarioDig,
    TpoPK idUniOrg,
    String codPtoTrab) throws TrException
```

Permite eliminar una firma digital de un documento.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento. Si el identificador es 'null', o no válido, o el expediente está archivado, se generará una excepción</i>
<i>String usuarioDig</i>	<i>Identificador del usuario digital</i>
<i>TpoPK idUniOrg</i>	<i>Identificador de la unidad orgánica u organismo.</i>

String codPtoTrab

Código del puesto de trabajo

2.4.1.4.2.6(w) Incluir firma digital a un documento

```
void incluirFirmaDigital(
    TpoPK idComponente,
    TpoPK idDocExp,
    String usuarioDig,
    TpoPK idUniOrg,
    String codPtoTrab,
    TpoDate fecha,
    String codTransaccion,
    byte[] pkcs7,
    String c_hash) throws TrException
```

Permite incluir una firma digital al documento que se indique en el parámetro idDocExp. Si se ha definido un componente para guardar los documentos se envían los datos de la firma.

El idComponente nos será útil posteriormente si queremos generar el documento en pdf y cifrar el código de transacción de la firma electrónica.

Entradas

<i>TpoPK idComponente</i>	<i>Identificador del componente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idDocExp</i>	<i>Identificador del documento. Si el identificador es 'null', o no válido, o el expediente está archivado, se generará una excepción</i>
<i>String usuarioDig</i>	<i>Identificador del usuario digital</i>
<i>TpoPK idUniOrg</i>	<i>Identificador de la unidad orgánica u organismo.</i>
<i>String codPtoTrab</i>	<i>Código del puesto de trabajo</i>
<i>TpoDate fecha</i>	<i>Fecha de la firma. Obligatoria.</i>
<i>String codTransaccion</i>	<i>Código de la transacción de firma</i>
<i>byte[] pkcs7</i>	<i>Recibo de la firma digital</i>
<i>String c_hash</i>	<i>Código hash de la firma digital</i>



void incluirFirmaDigital(

TpoPK idComponente,
TpoPK idDocExp,
String usuarioDig,
TpoPK idUniOrg,
String codPtoTrab,
TpoDate fecha,
String codTransaccion,
byte[] pkcs7) throws TrException

Deprecado

Entradas

<i>TpoPK idComponente</i>	<i>Identificador del componente Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idDocExp</i>	<i>Identificador del documento. Si el identificador es 'null', o no válido, o el expediente está archivado, se generará una excepción</i>
<i>String usuarioDig</i>	<i>Identificador del usuario digital</i>
<i>TpoPK idUniOrg</i>	<i>Identificador de la unidad orgánica u organismo.</i>
<i>String codPtoTrab</i>	<i>Código del puesto de trabajo</i>
<i>TpoDate fecha</i>	<i>Fecha de la firma. Obligatoria.</i>
<i>String codTransaccion</i>	<i>Código de la transacción de firma</i>
<i>byte[] pkcs7</i>	<i>Recibo de la firma digital</i>

2.4.1.4.2.7 Incluir firma digital documento delegado

void incluirFirmaDigitalDocumentoDelegado(

TpoPK idDocExp,
TpoDate fecha,
String codTransaccion,
byte[] pkcs7,
String c_hash,
String tipoFirmaEni) throws TrException

Permite firmar un documento que se pasa como parámetro.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento. Si el identificador es 'null', o no válido, o el expediente está archivado, se generará una excepción</i>
<i>TpoDate fecha</i>	<i>Fecha de la firma. Obligatoria.</i>
<i>String codTransaccion</i>	<i>Código de la transacción de firma</i>
<i>byte[] pkcs7</i>	<i>Recibo de la firma digital</i>
<i>String c_hash</i>	<i>Código hash asociado a la firma. Poner a null si no se inserta.</i>



String tipoFirmaEni

Tipo firma:

- TF01
- TF02
- TF03
- TF04
- TF05
- TF06

2.4.1.4.2.8 Filtro de firmantes

boolean isFiltroFirmantes()

Devuelve el valor del atributo filtroFirmantes que indica si se aplica el filtro al obtener los firmantes del tipo de documento.

Salidas

boolean filtroFirmantes

Indica si se aplica el filtro de firmantes
Por defecto su valor es true.

2.4.1.4.2.9 Firmantes definidos

TrFirmante[] obtenerFirmantesDefinidos(
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener todos los firmantes definidos en el sistema.

Entradas

ClausulaWhere where

Filtro a aplicar.
Consultar campos posibles para TrFirmante.

ClausulaOrderBy orderBy

Ordenación a aplicar.
Consultar campos posibles para TrFirmante.

Salida

TrFirmante []

Array de objetos TrFirmante.

2.4.1.4.2.10 Disposiciones firmantes

TrTextoDisposicion[] obtenerDisposicionesFirmante(
TpoPK idTeDi,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos de la disposición que se indica como parámetro.

Entradas

TpoPK idTeDi

Identificador de la disposición.
Si el identificador es "null" devolverá todas.

ClausulaWhere where

Filtro a aplicar.

Consultar campos posibles para TrTextoDisposicion.

ClausulaOrderBy orderBy

Ordenación a aplicar.

Consultar campos posibles para TrTextoDisposicion.

Salida

TrTextoDisposicion []

Array de objetos TrTextoDisposicion.

2.4.1.4.2.11 Obtener documento externo firmado

byte[] obtenerDocumentoExternoFirmado(

String idDocExpte
String motivo) throws TrException

Obtiene el documento con su cajetín de firma de un documento custodiado en un repositorio externo. El cajetín de firma contiene información de los firmantes como de la propia firma. Para ello se realiza una consulta a la Herramienta Centralizada de Verificación (HCV). En caso de no devolver informe de firma la HCV, se generará con la información contenida en el documento ENI devuelto por la HCV (contenido y firmas) siempre que el contenido sea modificable (no está securizado ni esté corrupto). En caso de estarlo, se devuelve el contenido original. Si se produce algún error la API lanza una excepción.

Entradas

String idDocExpte

Identificador del documento.

String motivo

Motivo para realizar la consulta al repositorio externo.

Salida

byte[]

Contenido binario del documento con su cajetín de firma con información de los firmantes.

2.4.1.4.2.12 Obtener documento firmado

byte[] obtenerDocumentoFirmado(

String idDocExpte) throws TrException

Obtiene el documento con su cajetín de firma. El cajetín de firma contiene información de los firmantes como de la propia firma.

Entradas

String idDocExpte

Identificador del documento.

Salida

byte[]

Contenido binario del documento con su cajetín de firma con información de los firmantes.

2.4.1.4.2.13 Obtener firmantes firma dinámica.

TrFirmante[] obtenerFirmantesFirmaDinamica(
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener todos los firmantes definidos en el sistema para la firma dinámica.

Entradas

<i>ClausulaWhere where</i>	Cláusula where para filtrar el resultado.
<i>ClausulaOrderBy orderBy</i>	Cláusula Order By para ordenar el resultado.

Salida

<i>TrFirmante[]</i>	Array de objetos TrFirmante
---------------------	-----------------------------

2.4.1.4.2.14 Obtener límites de relaciones entre interesados

TrLimiteRelacion[] obtenerLimitesRelacion(
 TpoPK idLimiteRela,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener un conjunto de límites para las relaciones entre interesados.

Entradas

<i>TpoPK idLimiteRela</i>	<i>Identificador del límite</i> Si el identificador es 'null', o no válido se devolverán todos
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para <i>TrLimiteRelacion</i> .
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para <i>TrLimiteRelacion</i> .

Salida

<i>TrLimiteRelacion []</i>	Array de objetos <i>TrLimiteRelacion</i>
-----------------------------	--

2.4.1.4.2.15 Insertar informe de firma

Void insertarInformeFirma(
 final String idDocExpte,
 final byte[] informeFirma) throws TrException

Insertar informe de firma a partir del identificador del documento expediente y el byte [] del informe

Entradas

String idDocExpte	<i>Identificador del documento expediente</i>
byte[] informeFirma	<i>Informe de firma</i>

Salida

Void

2.4.1.4.2.16 Generar informe de firma

Void insertarInformeFirma(final String idDocExpte) throws TrException

Insertar informe de firma a partir del identificador del documento expediente y el byte [] del informe

Entradas

String idDocExpte *Identificador del documento expediente*

Salida

Void

2.4.1.4.2.17 Recuperar informe firma documento expediente

Void recuperarInformeFirmaDocumentoExpediente(final TpoPK idDocExpte) throws TrException

Recupera el informe de firma de un documento expediente de BBDD

Entradas

TpoPK idDocExpte *Identificador del documento expediente*

Salida

byte [] *Array de byte*

2.4.1.4.2.18 Obtener informe firma documento expediente

Byte[] obtenerInformeFirmaDocumentoExpediente(final String idDocExpte) throws TrException

Genera informe de firma con registro al instante.

Entradas

String idDocExpte *Identificador del documento expediente*

Salida

Void *Array de byte*

2.4.1.4.2.19 Obtener documento cajetín

Byte[] obtenerDocumentoCajetin(
final String idDocExpte,
TrDocumentoExpediente documento) throws TrException

Genera informe de firma con registro al instante.

Entradas

String idDocExpte	<i>Identificador del documento expediente</i>
TrDocumentoExpediente documento	Clase que representa la información asociada a un documento generado y/o incorporado a un documento.

Salida

Byte []	<i>Array de byte</i>
---------	----------------------

2.4.1.4.3 APIs exclusivas para Expediente Electrónico

2.4.1.4.3.1 Importar documento ENI

TpoPK importarDocumentoEni (InputStream documentoXml, TpoPK idExp, TpoPK idExpFase, TpoPK idTipoDoc) throws TrException

Importar un expediente desde la información ENI.

Entrada

InputStream documentoXml	Xml documento electrónico
<i>TpoPK idExp</i>	Identificador del expediente
<i>TpoPK idExpFase</i>	Identificador del expediente en fase
TpoPK idTipDoc	Identificador del tipo de documento

Salida

<i>TpoPK</i>	Identificador del documento ENI importado
--------------	---

2.4.1.4.3.2 Exportar expediente ENI

File exportarExpedienteENI (TpoPK idExpte, boolean documentos, boolean pdfResumen, boolean infoAuxPdfResumen, TpoPK refInteresado, String motivo) throws TrException

Genera un zip con la información del expediente electrónico y su índice firmado, así como todos los documentos electrónicos firmados del expediente si así se indica. Devuelve un fichero zip que empaqueta el xml de expediente y los documentos. Los documentos existentes en el tramitador serán actualizados con el identificador de documento generado para el índice, al igual que el expediente. Si se pasa la referencia del interesado, se adjuntan solo los documentos filtrados por el interesado.

Entrada

<i>TpoPK idExpte</i>	Identificador de expediente
<i>boolean documentos</i>	Indica si se quiere exportar el expediente junto con los documentos. Si no se indica, el método devuelve un fichero xml, si se indica, devuelve un zip que contiene el xml del índice del expediente y los xml's de los documentos ordenados por carpetas.
<i>boolean pdfResumen</i>	Indica si se genera el pdf resumen del expediente para mostrar en la visualización del índice
<i>boolean infoAuxPdfResumen</i>	Indica si se incluye en el pdf resumen información auxiliar de los documentos.
<i>TpoPK refInteresado</i>	Identificador de interesado. Si se indica se adjuntan únicamente los documentos filtrados por el interesado.
<i>String motivo</i>	Motivo para obtener los documentos de repositorios externos en caso que aplique.

Salida

<i>File</i>	Fichero con el contenido del expediente electrónico.
-------------	--

<p>File exportarExpedienteENI (<i>TpoPK idExpte</i>, <i>boolean documentos</i>, <i>boolean pdfResumen</i>, <i>boolean infoAuxPdfResumen</i>, <i>TpoPK refInteresado</i>) throws <i>TrException</i></p>	<i>Deprecado</i>
---	------------------

Entrada

<i>TpoPK idExpte</i>	Identificador de expediente
<i>boolean documentos</i>	Indica si se quiere exportar el expediente junto con los documentos. Si no se indica, el método devuelve un fichero xml, si se indica, devuelve un zip que contiene el xml del índice del expediente y los xml's de los documentos ordenados por carpetas.

<code>boolean pdfResumen</code>	Indica si se genera el pdf resumen del expediente para mostrar en la visualización del índice
<code>boolean infoAuxPdfResumen</code>	Indica si se incluye en el pdf resumen información auxiliar de los documentos.
<code>TpoPK refInteresado</code>	Identificador de interesado. Si se indica se adjuntan únicamente los documentos filtrados por el interesado.

Salida

<code>File</code>	Fichero con el contenido del expediente electrónico.
-------------------	--

<p>String exportarExpedienteENI (TpoPK idExpte, boolean documentos, boolean pdfResumen, boolean infoAuxPdfResumen, TpoPK refInteresado) throws TrException</p>	<i>Deprecado</i>
---	------------------

Entrada

<code>TpoPK idExpte</code>	Identificador de expediente
<code>OutputStream xml</code>	Fichero xml del expediente electrónico (E/S), firma el índice de expediente.

Salida

<code>File</code>	Fichero con el contenido del expediente electrónico.
-------------------	--

2.4.1.4.3.3 Generador numerador ENI

String generarNumeradorEni () throws TrException
--

2.4.1.4.3.4 Importar expediente ENI

<p>TpoPK importarExpedienteEni (InputStream expedienteXml, TpoPK idUniOrg, TpoPK idDefProc, TpoPK idTipoExpte, String titulo, String observaciones) throws TrException</p>

Importar un expediente desde la información ENI.

Entrada

<code>InputStream expedienteXml</code>	Xml Expediente electronico
--	----------------------------

<i>TpoPK idUniOrg</i>	Identificador de la unidad orgánica
<i>TpoPK idDeProc</i>	Identificador del procedimiento
<i>TpoPK idTipoExpte</i>	Identificador del tipo de expediente
String titulo	Título del expediente
String observaciones	Observaciones

Salida

<i>TpoPK</i>	Identificador del expediente ENI importado
--------------	--

2.4.1.4.3.5 Obtener documento ENI por referencia

<p>DocumentoEni obtenerDocumentoEniPorReferenciaCSV(String csv, TpoString motivo) throws TrException</p>
--

Obtiene un documento ENI a partir del CSV

Entrada

String csv	CSV
<i>TpoString motivo</i>	Motivo de la obtención del documento

Salida

<i>DocumentoEni</i>	Documento ENI
---------------------	---------------

2.4.1.4.3.6 Obtener estado elaboración del documento ENI

<p>TrEstadoElaboracionDocEni[] obtenerEstadoElaboracionDocEni(TpoPK idEstelab, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Obtiene el estado de elaboración del documento ENI.

Entrada

TpoPK idEstelab	Identificador de estado de elaboración del documento ENI
<i>ClausulaWhere where</i>	Cláusula where
<i>ClausulaOrderBy orderBy</i>	Cláusula order by

Salida

<i>DocumentoEni</i>	Documento ENI
---------------------	---------------

2.4.1.4.3.7 Obtener identificador ENI documento

TrIdentificadorEni obtenerIdentificadorEniDocumento (
TpoPK idDocumento) throws TrException

Obtiene el identificador normalizado del documento electrónico adaptado al ENI.

Entrada

<i>TpoPK idDocumento</i>	Identificador del documento en Trew@.
--------------------------	---------------------------------------

2.4.1.4.3.8 Obtener identificador ENI expediente

TrIdentificadorEni obtenerIdentificadorEniExpediente (
TpoPK idExpediente) throws TrException

Obtiene el identificador normalizado del expediente electrónico adaptado al ENI.

Entrada

<i>TpoPK idExpediente</i>	Identificador del expediente en Trew@.
---------------------------	--

2.4.1.4.3.9 Actualización estado documento ENI

void actualizarEstadoElaboracionDocEni (
TpoPK idDocExp,
TpoPK idEstelab) throws TrException

Actualiza el estado de elaboración del documento ENI.

Entrada

<i>TpoPK idDocExp</i>	Identificador del documento expediente en Trew@.
<i>TpoPK idEstelab</i>	Identificador del estado de elaboración.

2.4.1.4.3.10 Construye un expediente ENI

```
Map<byte[], List<TrDocumentoExpediente>> construirExpEniexpWS(  

    final TpoPK idExpte,  

    final TpoPK idInt) throws TrException
```

Construye un expediente ENI

Entrada

<i>TpoPK idExpte</i>	Identificador del expediente en Trew@.
<i>TpoPK idInt</i>	Identificador del interesado.

2.4.1.4.3.11 Eliminar notificación de un interesado

```
void eliminarNotificacionInteresado(  

    TpoPK idDocExp,  

    TpoPK idExpediente,  

    TpoPK idInteresado,  

    TpoPK idRazonInt,  

    TpoPK idRazonIntDocu) throws TrException
```

Permite eliminar una notificación de un interesado.

Entrada

<i>TpoPK idDocExp</i>	Identificador del expediente en Trew@.
<i>TpoPK idExpediente</i>	Identificador del interesado.
<i>TpoPK idInteresado</i>	Identificador del interesado
<i>TpoPK idRazonInt</i>	Identificador de la razón de interés

TpoPK idRazonIntDocu

Identificador de la razón de interés del documento. Si se pasa null no se filtra por la razón de interés.

2.4.1.4.3.12 Eliminación del acceso a libreoffice

```
void eliminarAccesoLibreoffice(
    TpoPK idAcli) throws TrException
```

Elimina el acceso a libreoffice.

Entrada

TpoPK idAcli

Identificador del acceso.

2.4.1.4.3.13 Inserta un nuevo acceso a libreoffice

```
void insertarAccesoLibreoffice(
    TrAccesoLibreoffice accesoLibreoffice) throws TrException
```

Inserta un nuevo acceso a libreoffice.

Entrada

TrAccesoLibreoffice
accesoLibreoffice

Acceso de libreoffice.

2.4.1.4.3.14 Modificar acceso de libreoffice

TpoPK modificarAccesoLibreoffice(
TrAccesoLibreoffice accesoLibreoffice) throws TrException

Modifica un acceso de libreoffice.

Entrada

TrAccesoLibreoffice accesoLibreoffice	Acceso de libreoffice.
--	------------------------

2.4.1.4.3.15 Modificar notificación de un interesado

void modificarNotificacionInteresado(
TrNotificacionInteresado notificacionInte) throws TrException

Permite modificar una notificación de envío para un interesado en un documento.

Entrada

TrNotificacionInteresado notificacionInte	Notificación a modificar.
--	---------------------------

2.4.1.4.3.16 Obtiene los accesos a libreoffice

TrAccesoLibreoffice[] obtenerAccesoLibreoffice(
TpoPK idAcceso,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Obtiene los accesos de libreoffice.

Entrada

TpoPK idAcceso	Identificador del acceso
ClausulaWhere where	Clausula where
ClausulaOrderBy orderBy	Clausula order

2.4.1.4.3.17 Obtiene los datos registro del documento

<p>TrRegistroDocumento obtenerDatosRegistroDocumento(TpoPK idDocExp) throws TrException <i>Deprecado</i></p>

Devuelve los datos de registro que existen para un documento que se indica como parámetro.

Entrada

TpoPK idDocExp	Identificador del documento que se quieren los datos del registro
----------------	---

2.4.1.4.3.18 Obtener interesados de un expediente dado de baja

<p>TrInteresadoExpediente[] obtenerInteresadosExpedienteDadoDeBaja(TpoPK idExp, TpoPK idInte, TpoPK idRazonInt, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Permite obtener un conjunto de interesados dados de baja en un expediente.

Entrada

TpoPK idExp	Identificador del expediente, si no se indica se obtienen los de todos los expedientes
TpoPK idInte	Identificador del interesado, si no se indica se obtienen todos.
TpoPK idRazonInt	Identificador de la razón de interés. Si se pasa null no se filtra por la razón de interés.
ClausulaWhere where	Cláusula where para filtrar el resultado
ClausulaOrderBy orderBy	Clausula order para ordenar el resultado

2.4.1.4.3.19 Obtiene los organismos por perfil de usuarios

<p>TpoPK[] obtenerOrganismosPorPerfilUsuario(String perfil) throws TrException</p>
--

Obtiene los identificadores de organismos que poseen puestos de trabajos con empleados vigentes que tengan el perfil especificado mediante parámetro, es decir, obtiene los identificadores de organismos para los que existe un empleado cuyo perfil sea el especificado.

Entrada

String perfil	Perfil de usuarios
---------------	--------------------

2.4.1.4.3.20Obtiene las notificaciones de los interesados

<p>TrNotificacionInteresado[] obtenerNotificacionesInteresado(TpoPK idExpediente, TpoPK idInteresado, TpoPK idRazonInt, TpoPK idDocExp, TpoPK idRazonIntDocu, ClausulaWhere where, ClausulaOrderBy orderby) throws TrException <i>Deprecado</i></p>
--

Permite obtener las notificaciones de los interesados en los documentos de los expedientes.

Entrada

TpoPK idExpediente	Identificador del expediente, si no se indica se obtienen las de todos los expedientes.
TpoPK idInteresado	Identificador del interesado, si no se indica se obtienen los de todos los interesados
TpoPK idRazonInt	Identificador de la razón de interés. Si se pasa null no se filtra por la razón de interés.
TpoPK idDocExp	Identificador del documento del expediente. Si se le pasa null no se filtra por el documento del expediente.
TpoPK idRazonIntDocu	Identificador de la razón de interés del documento. Si se pasa null no se filtra por la razón de interés
ClausulaWhere where	Clausula where para filtrar el resultado

ClausulaOrderBy orderBy Clausula order para odernar el resultado.

2.4.1.4.4 APIs sobre expedientes

2.4.1.4.4.1(w) Archivo de un expediente

```
void archivarExpediente(
    TpoPK idExpediente,
    TpoDate fecha) throws TrException
```

Permite archivar un expediente siempre que no esté archivado. Para ello se comprueba que el identificador del expediente sea válido y no sea un expediente ya archivado, generando una excepción en caso contrario. Además archiva todos los documentos archivables del expediente.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null' se generará una excepción.
<i>TpoDate fecha</i>	<i>Fecha de archivo del expediente</i> Si no se indica fecha de archivo, por defecto se tomará la del sistema.

2.4.1.4.4.2 Anular el archivo de un expediente

```
void anularArchivoExpediente (
    TpoPK idExpediente) throws TrException
```

Anula el archivo de un expediente siempre que esté archivado. Para ello se comprueba que el identificador del expediente sea válido y no sea un expediente que no está archivado, generando una excepción en caso contrario. Además anula la fecha de archivo de todos los documentos archivables del expediente.

Entradas

TpoPK IdExpediente

Identificador del expediente.

Si el identificador es 'null' se generará una excepción.

2.4.1.4.4.3 Consultar estado de petición de Archivo de un expediente

```
void consultarArchivado (  
    TpoPK idExpediente) throws TrException
```

Realiza una consulta al sistema externo de archivo @rchivA para actualizar el estado del expediente. Debe existir un componente @RCHIVA definido en el sistema y previamente debe haberse realizado una solicitud de ingreso del expediente en @rchivA. En caso de que la solicitud de ingreso del expediente sea satisfactoria, se actualizará la fecha de archivo del expediente, y de todos sus documentos archivables, así como el estado y número de registro del expediente.

Entradas

TpoPK IdExpediente

Identificador del expediente.

Si el identificador es 'null' se generará una excepción.

2.4.1.4.4.4 Alta de expedientes

```
TpoPK crearExpediente(  
    TpoPK idTipoExp,  
    TpoPK idDefProc,  
    TpoDate fechaAlta,  
    String numExp,  
    String tituloExp,  
    String observaciones,  
    TpoPK idUniOrg,  
    TpoPK idUniOrgEnvia,  
    String urlWanda  
    ) throws TrException
```

Método que da de alta un nuevo expediente para tramitar en Trew@, para un tipo de expediente dado y/o una definición de procedimiento y una fecha de alta del expediente. Al expediente se le asignará la versión del procedimiento que corresponde a la fecha de alta suministrada. Se debe indicar como mínimo uno de los dos identificadores, el del tipo de expediente o el de la definición del procedimiento.

Entradas

TpoPK idTipoExp

Identificador del tipo de expediente.

Si no se indica identificador del tipo de expediente se deberá indicar el identificador del procedimiento.

TpoPK idDefProc

Identificador del procedimiento a seguir.

Si no se indica identificador del procedimiento se deberá indicar el identificador del tipo de expediente.

<i>TpoDate fechaAlta</i>	<i>Fecha de alta.</i>
	<i>Entrada/Salida.</i>
<i>String numExp</i>	<i>Si no se indica fecha de alta, por defecto se tomará la del sistema. Número del expediente para la aplicación cliente</i>
<i>String tituloExp</i>	<i>Título del expediente para la aplicación cliente</i>
<i>String observaciones</i>	<i>Observaciones para el expediente</i>
<i>TpoPK idUniOrg</i>	<i>Identificador de la unidad orgánica u organismo a la que pertenece el expediente</i>
<i>TpoPK idUniOrgEnvia</i>	<i>Identificador de la unidad orgánica u organismo que envía el expediente</i>
<i>String urlWanda</i>	<i>Url del expediente en w@ndA</i>

Salida

<i>TpoPK</i>	<i>Identificador del nuevo expediente.</i>
--------------	--

Se ofrece la posibilidad de indicar el procedimiento a seguir por el expediente (aunque no sea la que le corresponda según la fecha de vigencia de la versión) siempre que exista una versión para el tipo de expediente que siga ese procedimiento.

El método devuelve el identificador único con el que Trew@ identifica el expediente, que será imprescindible para la mayoría de funciones, es decir, los sistemas deben definir clave ajena a la entidad de expedientes de Trew@. También se devuelve la fecha con la que ha sido dado de alta el expediente en Trew@.

Además en la nueva versión permite indicar la url del expediente en w@ndA.

A partir de 1.6.0 existe la posibilidad de indicar el usuario propietario del expediente mediante el usuario establecido en el TrAPIUI mediante el api establecerUsuarioSistema.

TpoPK crearExpediente(

TpoPK idTipoExp
TpoPK idDefProc,
TpoDate fechaAlta) throws TrException

Deprecado

Entradas

<i>TpoPK idTipoExp</i>	<i>Identificador del tipo de expediente.</i>
	<i>Si no se indica identificador del tipo de expediente se deberá indicar el identificador del procedimiento.</i>
<i>TpoPK idDefProc</i>	<i>Identificador del procedimiento a seguir.</i>
	<i>Si no se indica identificador del procedimiento se deberá indicar el identificador del tipo de expediente.</i>

TpoDate fechaAlta

Fecha de alta.

Entrada/Salida.

Si no se indica fecha de alta, por defecto se tomará la del sistema.

Salida

TpoPK

Identificador del nuevo expediente.

TpoPK crearExpediente(
 TpoPK idTipoExp
 TpoPK idDefProc,
 TpoDate fechaAlta,
 String numExp,
 String tituloExp,
 String observaciones,
 TpoPK idUniOrg,
 TpoPK idUniOrgEnvia) throws TrException

Entradas

TpoPK idTipoExp

Identificador del tipo de expediente.

Si no se indica identificador del tipo de expediente se deberá indicar el identificador del procedimiento.

TpoPK idDefProc

Identificador del procedimiento a seguir.

Si no se indica identificador del procedimiento se deberá indicar el identificador del tipo de expediente.

TpoDate fechaAlta

Fecha de alta.

Entrada/Salida.

Si no se indica fecha de alta, por defecto se tomará la del sistema.

String numExp

Número del expediente para la aplicación cliente

String tituloExp

Título del expediente para la aplicación cliente

String observaciones

Observaciones para el expediente

TpoPK idUniOrg

Identificador de la unidad orgánica u organismo a la que pertenece el expediente

TpoPK idUniOrgEnvia

Identificador de la unidad orgánica u organismo que envía el expediente

Salida

TpoPK

Identificador del nuevo expediente.

TpoPK crearExpediente(
 TpoPK idTipoExp

TpoPK idDefProc,
TpoDate fechaAlta,
String numExp,
String tituloExp,
String observaciones,
TpoPK idUniOrg,
TpoPK idUniOrgEnvia,
String urlWanda,
boolean iniciadoCiudadano) throws TrException

Entradas

<i>TpoPK idTipoExp</i>	<i>Identificador del tipo de expediente.</i>
	Si no se indica identificador del tipo de expediente se deberá indicar el identificador del procedimiento.
<i>TpoPK idDefProc</i>	<i>Identificador del procedimiento a seguir.</i>
	Si no se indica identificador del procedimiento se deberá indicar el identificador del tipo de expediente.
<i>TpoDate fechaAlta</i>	<i>Fecha de alta.</i>
	Entrada/Salida.
String numExp	Si no se indica fecha de alta, por defecto se tomará la del sistema. <i>Número del expediente para la aplicación cliente</i>
String tituloExp	<i>Título del expediente para la aplicación cliente</i>
String observaciones	<i>Observaciones para el expediente</i>
TpoPK idUniOrg	<i>Identificador de la unidad orgánica u organismo a la que pertenece el expediente</i>
TpoPK idUniOrgEnvia	<i>Identificador de la unidad orgánica u organismo que envía el expediente</i>
String urlWanda	<i>Url del expediente en w@ndA</i>
Boolean iniciadoCiudadano	<i>Booleano que indica si el expediente ha sido iniciado por el ciudadano</i>

Salida

<i>TpoPK</i>	<i>Identificador del nuevo expediente.</i>
--------------	--

2.4.1.4.4.5 Crear expediente hijo

<p>TpoPK crearExpediente(TpoPK idTipoExp, TpoPK idDefProc, TpoDate fechaAlta, String numExp, String tituloExp, String observaciones, TpoPK idUniOrg, TpoPK idUniOrgEnvia, String urlWanda, boolean ini_Ciudadano, String identificadorPadre, String abreviaturaRazonInteres, TpoDate fechaCreacionEntrega) throws TrException</p>
--

Permite crear expedientes hijos relacionados con el expediente padre proporcionado a partir de los parámetros definidos

Entradas

TpoPK idTipoExp

Identificador del tipo de expediente.

Si no se indica identificador del tipo de expediente se deberá indicar el identificador del procedimiento.

TpoPK idDefProc

Identificador del procedimiento a seguir.

Si no se indica identificador del procedimiento se deberá indicar el identificador del tipo de expediente.

<i>TpoDate fechaAlta</i>	<i>Fecha de alta.</i>
	<i>Entrada/Salida.</i>
String numExp	<i>Si no se indica fecha de alta, por defecto se tomará la del sistema. Número del expediente para la aplicación cliente</i>
String tituloExp	<i>Título del expediente para la aplicación cliente</i>
String observaciones	<i>Observaciones para el expediente</i>
TpoPK idUniOrg	<i>Identificador de la unidad orgánica u organismo a la que pertenece el expediente</i>
TpoPK idUniOrgEnvia	<i>Identificador de la unidad orgánica u organismo que envía el expediente</i>
String urlWanda	<i>Url del expediente en w@ndA</i>
Boolean ini_Ciudadano	<i>Booleano que indica si el expediente ha sido iniciado por el ciudadano</i>
String identificadorPadre	<i>Identificador ENI o CSV del expediente padre</i>
String abreviaturaRazonInteres	<i>Abreviatura de la razón de interés</i>
String nombreTransicion	<i>Nombre de la transición del expediente que se realiza en VeA</i>
TpoDate fechaLimite	<i>Fecha límite</i>
TpoDate fechaCreacionEntrega	<i>Fecha de la creación de la entrega</i>
<u>Salida</u>	
<i>TpoPK</i>	<i>Identificador del nuevo expediente.</i>

2.4.1.4.4.6 Obtención de datos del expediente

TrExpediente obtenerDatosExpediente(
TpoPK idExpediente) throws TrException

Devuelve para un expediente dado, los datos del expediente guardados en TREW@. Si no se indica identificador del expediente el método lanzará una excepción.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null', se generará una excepción
---------------------------	--

Salida

<i>TrExpediente</i>	<i>Objeto TrExpediente con los datos del expediente</i>
---------------------	---

TrExpediente obtenerDatosExpediente(
TpoPK idExpediente,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null', se generará una excepción
---------------------------	--

<i>ClausulaWhere where</i>	<i>Cláusula where para filtrar el resultado</i>
----------------------------	---

ClausulaOrderBy orderBy

Cláusula order para ordenar el resultado

Salida

TrExpediente

Objeto TrExpediente con los datos del expediente

2.4.1.4.4.7 Obtención de relaciones entre expedientes

```
TrRelacionExpediente[ ] obtenerRelacionesExpediente(
    TpoPK idExpediente,
    ClausulaWhere where,
    ClausulaOrderBy orderBy) throws TrException
```

Permite obtener las relaciones del expediente que se pasa por parámetros con otros expedientes. Devuelve tanto las relaciones en las que el expediente sea maestro como detalle de otros.

El tipo de relación entre expedientes puede ser “M” – Maestro, “D” – Detalle o “I” – Igualdad. Si se indica el parámetro idExpediente se tomará como el id de referencia para interpretar los resultados.

Por ejemplo si en base de datos existen los siguientes datos:

Id relación	Id expediente	Tipo relación	Id expediente relacionado
1	171155	M	173612
2	171155	D	173592

Éstos datos se podrían representar de la siguiente forma:



Con estos datos si se realiza una consulta indicando el parámetro idExpediente con el valor 171155 obtendremos 2 resultados de TrRelacionExpediente cuyos valores serán:

REFRELACIONEXP	REFEXPEDIENTE	TIPORELACION
1	173612	M
2	179532	D

Podemos comprobar que el expediente 171155 es Maestro del 173612 y es detalle del 179532. La forma correcta de interpretar los resultados sería a partir del idExpediente consultado. El "idExpediente" es "TIPORELACION" de "REFEXPEDIENTE". Por ejemplo el 171155 es Maestro del 173612.

Otro ejemplo, si realizamos la consulta indicando el parámetro idExpediente con el valor 173612 obtendríamos 1 resultado de TrRelacionExpediente con los siguientes valores:

REFRELACIONEXP	REFEXPEDIENTE	TIPORELACION
1	171155	D

Su interpretación sería: el expediente 173612 es Detalle del expediente 171155.

Entradas

TpoPK IdExpediente

Identificador del expediente.

Si el identificador es 'null' se generará una excepción.

ClausulaWhere where

Filtro a aplicar.

Consultar campos posibles para TrCambioProcedimientoExpediente.

ClausulaOrderBy orderBy

Ordenación a aplicar.

Consultar campos posibles para TrCambioProcedimientoExpediente.

Salida

TrRelacionExpediente []

Array de objetos TrRelacionExpediente.

2.4.1.4.4.8Relacionar expediente padre e hijos

List<BigDecimal> relacionarExpedientePorENI(
 String eniPadre,
 List<String> eniHijos) throws TrException

Relaciona los expedientes a partir de un ENI del padre con los identificadores de los expedientes hijos.

Entradas

String eniPadre

Identificador ENI o CSV del expediente padre.

List<String> eniHijos

Lista de identificadores ENI de los expedientes hijos.

2.4.1.4.4.9 Obtención de expedientes

<p>TrExpediente[] obtenerExpedientes(TpoPK idExpediente, ClausulaWhere where, ClausulaOrderBy orderBy, ClausulaExclusion exclusiones) throws TrException;</p>
--

Permite obtener los datos de un expediente. Si se le pasa el parámetro idExpediente a null se obtienen todos los expedientes.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null' se obtienen todos los expedientes,
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrExpediente.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrExpediente.
<i>ClausulaExclusion exclusiones</i>	<i>Entidades a excluir de la consulta.</i> Consultar exclusiones posibles para TrExpediente

Salida

<i>TrExpediente []</i>	<i>Array de objetos TrExpediente.</i>
-------------------------	---------------------------------------

<p>TrExpediente[] obtenerExpedientes(TpoPK idExpediente, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException;</p>
--

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null' se obtienen todos los expedientes,
---------------------------	--

ClausulaWhere where

Filtro a aplicar.

Consultar campos posibles para TrExpediente.

ClausulaOrderBy orderBy

Ordenación a aplicar.

Consultar campos posibles para TrExpediente.

Salida

TrExpediente []

Array de objetos TrExpediente.

2.4.1.4.4.10 Eliminación de usuarios asignados a un expediente

```
void eliminarUsuarioAsignadosExpediente (
    TpoPK idExpediente,
    String usuario) throws TrException
```

Permite eliminar un usuario asignado a un expediente. Si el parámetro usuario es 'null' se eliminan todos los usuarios asignados al expediente.

Entradas

TpoPK IdExpediente

Identificador del expediente.

Si el identificador es 'null' se generará una excepción.

String usuario

Usuario

Si el identificador es 'null' se eliminan todos los usuarios asignados al expediente.

```
void eliminarUsuarioAsignadosExpediente (
    TpoPK idUsuarioAsignado) throws TrException
```

Entradas

TpoPK IdUsuarioAsignado

Identificador del usuario asignado al expediente.

2.4.1.4.4.11 Eliminar organismo del expediente

```
void eliminarOrganismoExpediente( TpoPK idOrgExp) throws TrException;
```

Permite eliminar la asociación entre un organismo y un expediente.

Entradas

TpoPK idOrgExp

Identificador de la nueva asociación entre el organismo y el expediente. Si el identificador es 'null' o no válido se generará una excepción.

2.4.1.4.4.12 Eliminación de relaciones entre expedientes

```
void eliminarRelacionesExpediente (  
    TpoPK idExpediente,  
    TpoPK idRelacion) throws TrException
```

Permite eliminar una relación entre expedientes. Si no se indica *idRelacion* se eliminarán todas las relaciones del expediente.

Entradas

TpoPK idExpediente

*Identificador del expediente.
Si el identificador es 'null' se generará una excepción.*

TpoPK idRelacion

*Identificador de la relación entre expedientes
Si el identificador es 'null' se eliminarán todas las relaciones del expediente.*

2.4.1.4.4.13(b) Eliminación de expedientes

```
void eliminarExpediente(  
    TpoPK idExpediente ) throws TrException
```

Permite eliminar el expediente que se indica por parámetro, en Trew@. Está orientada a usuarios administradores. Si el sistema establecido tiene conexión al bus se informará del borrado del expediente siempre que el expediente haya sido previamente informado al bus w@ndA

Entradas

TpoPK idExpediente

Identificador del expediente a eliminar

2.4.1.4.4.14 Eliminar pago del expediente

```
void eliminarPagoTasa( TpoPK idPagoTasa) throws TrException;
```

Permite eliminar el pago de tasa de un expediente.

tualizados con el identificador de documento generado para el índice, al igual que el expediente. Si se pasa la referencia del interesado, se adjuntan solo los documentos filtrados por el interesado.

Entradas

<i>TpoPK idExpte</i>	<i>Identificador del expediente</i>
Boolean documentos	Indica si con documentos
Boolean pdfResumen	Indica si con pdf resumen
Boolean infoAuxPdfResumen	Indica si la información auxiliar del pdf resumen
<i>TpoPK refInteresado</i>	Identificador del interesado
String motivo	El motivo
Boolean admon	Indica si se realiza la llamada desde la Administración de Trew@
StringBuffer errorRecuperarDocExt	Indica si existe error al exportar documento externos (E/S)

2.4.1.4.4.17 Exploración de expedientes ordenados

```

TrExplorador[ ] explorarExpedientes(
    TpoPK idSistema,
    TpoPK idTipoExp,
    TpoPK idDefProc,
    TpoDate fechaAltaIni,
    TpoDate fechaAltaFin,
    String soloVigentes,
    String ordenarPor,
    String pendientes,
    String caducados,
    String reservados) throws TrException

```

Permite obtener un conjunto de expedientes, dependiendo de los filtros que se indiquen como parámetros, si el atributo *comprobarBloqueo* es false se devolverán todos los expedientes sin tener en cuenta si están bloqueados o no.

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema sobre el que se van a obtener los expedientes.</i> Si no se indica, de devolverán expedientes de todos los sistemas que cumplan el resto de filtros.
<i>TpoPK idTipoExp</i>	<i>Identificador del tipo de expediente.</i> Permite obtener sólo los expedientes de un tipo concreto
<i>TpoPK idDefProc</i>	<i>Identificador del procedimiento.</i> Permite obtener sólo los expedientes de un procedimiento concreto
<i>TpoDate fechaAltaIni</i>	<i>Fecha de alta inicial</i> Permite obtener sólo aquellos expedientes que fueron dados de alta después de la fecha indicada

<i>TpoDate fechaAltaFin</i>	<i>Fecha de alta fin</i> Permite obtener sólo aquellos expedientes que fueron dados de alta antes de la fecha indicada
<i>String soloVigentes</i>	<i>Permite obtener sólo los expedientes que son de un tipo de expediente vigente en el motor.</i> Por defecto 'N' → "S" – Sí → "N" – No
<i>String ordenarPor</i>	<i>Permite ordenar el resultado.</i> Por defecto 'T'. → "T" – Sistema, Tipo de expediente, Procedimiento, Metafase, Fase, Año, Expediente. → "E" – Sistema, Metafase, Fase, Tipo de expediente, Procedimiento, Año, Expediente. → "TEX" – Sistema, Tipo de expediente, Metafase, Fase, Año, Expediente. → "TEV" – Sistema, Procedimiento, Metafase, Fase, Año, Expediente. → "EEX" – Sistema, Metafase, Fase, Tipo de expediente, Año, Expediente. → "EEV" – Sistema, Metafase, Fase, Procedimiento, Año, Expediente. → "TO" – Sistema, Tipo de expediente, Procedimiento, Orden metafase, Metafase, Orden fase, Fase, Año, Expediente. → "EO" – Sistema, Orden metafase, Metafase, Orden fase, Fase, Tipo de expediente, Procedimiento, Año, Expediente.
<i>String pendientes</i>	<i>Permite obtener sólo los expedientes sobre los que el usuario tiene algún permiso para realizar tareas o cambios de fase.</i> Por defecto 'N'. → "S" – Sí → "N" – No
<i>String caducados</i>	<i>Permite obtener sólo los expedientes que se encuentran caducados.</i> Por defecto 'N'. → "S" – Sí → "N" – No
<i>String reservados</i>	<i>Permite obtener sólo los expedientes que se encuentran reservados por el usuario.</i> Por defecto 'N'. → "S" – Sí → "N" – No

Salida

TrExplorador [] *Array de objetos TrExplorador.*

<p>TrExplorador[] explorarExpedientes(TpoPK idSistema, String usuario, ClausulaWhere where, ClausulaOrderBy orderBy, ClausulaExclusion exclusions, boolean informacion, boolean informacionUsuario) throws TrException</p>
--

Permite obtener un conjunto de expedientes en su situación o situaciones actuales, dependiendo de los filtros que se indiquen como parámetros.

Se permite indicar mediante el campo TrExplorador.MAX_ROW en la ClausulaWhere el número máximo de registros a devolver en la consulta.

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema sobre el que se van a obtener los expedientes. Si no se indica, de devolverán expedientes de todos los sistemas que cumplan el resto de filtros.</i>
<i>String usuario</i>	<i>Identificador de usuario. Código del usuario para filtrar por los expedientes que tiene pendientes o en los que participa, opcional, si no se indica se cogerá el que está establecido en el TrAPIUI.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrExplorador</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrExplorador</i>
<i>ClausulaExclusio exclusiones</i>	<i>Entidades a excluir de la consulta. Consultar exclusiones posibles para TrExplorador</i>
<i>boolean informacion</i>	<i>Muestra información adicional sobre documentos, tareas y transiciones.</i>
<i>boolean informacionUsuario</i>	<i>Muestra información adicional sobre usuarios interesados y asignados al expediente.</i>

Salida

TrExplorador [] *Array de objetos TrExplorador.*

TrExplorador[] explorarExpedientes(
 TpoPK idSistema,
 String usuario,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener un conjunto de expedientes en sus situaciones actuales.

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema sobre el que se van a obtener los expedientes. Si no se indica, de devolverán expedientes de todos los sistemas que cumplan el resto de filtros.</i>
<i>String usuario</i>	<i>Identificador de usuario. Código del usuario para filtrar por los expedientes que tiene pendientes o en los que participa, opcional, si no se indica se cogerá el que está establecido en el TrAPIUI.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrExplorador</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrExplorador</i>

Salida

TrExplorador [] *Array de objetos TrExplorador.*

TrExplorador[] explorarExpedientes(
 TpoPK idSistema,
 String usuario,
 ClausulaWhere where,
 ClausulaOrderBy orderBy,
 ClausulaExclusion exclusiones) throws TrException

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema sobre el que se van a obtener los expedientes. Si no se indica, de devolverán expedientes de todos los sistemas que cumplan el resto de filtros.</i>
<i>String usuario</i>	<i>Identificador de usuario. Código del usuario para filtrar por los expedientes que tiene pendientes o en los que participa, opcional, si no se indica se cogerá el que está establecido en el TrAPIUI.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrExplorador</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrExplorador</i>
<i>ClausulaExclusion exclusiones</i>	<i>Entidades a excluir de la consulta. Consultar exclusiones posibles para TrExplorador</i>

Salida

TrExplorador [] *Array de objetos TrExplorador.*

TrExplorador[] explorarExpedientes(
 TpoPK idSistema,
 String usuario,
 ClausulaWhere where,
 ClausulaOrderBy orderBy,
 ClausulaExclusion exclusiones,
 boolean informacion) throws TrException

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema sobre el que se van a obtener los expedientes. Si no se indica, de devolverán expedientes de todos los sistemas que cumplan el resto de filtros.</i>
<i>String usuario</i>	<i>Identificador de usuario. Código del usuario para filtrar por los expedientes que tiene pendientes o en los que participa, opcional, si no se indica se cogerá el que está establecido en el TrAPIUI.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrExplorador</i>

<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrExplorador</i>
<i>ClausulaExclusion exclusiones</i>	<i>Entidades a excluir de la consulta. Consultar exclusiones posibles para TrExplorador</i>
<i>Boolean informacion</i>	<i>Muestra información adicional sobre documentos, tareas y transiciones</i>

Salida

TrExplorador [] *Array de objetos TrExplorador.*

TrExplorador[] explorarExpedientes(
 TpoPK idSistema,
 String usuario,
 TpoPK idTipoExp,
 TpoPK idDefProc,
 TpoDate fechaAltaIni,
 TpoDate fechaAltaFin,
 String soloVigentes,
 String ordenarPor,
 String pendientes,
 String caducados,
 String reservados) throws TrException

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema sobre el que se van a obtener los expedientes. Si no se indica, de devolverán expedientes de todos los sistemas que cumplan el resto de filtros.</i>
<i>String usuario</i>	<i>Usuario</i>
<i>TpoPK idTipoExp</i>	<i>Identificador del tipo de expediente. Permite obtener sólo los expedientes de un tipo concreto</i>
<i>TpoPK idDefProc</i>	<i>Identificador del procedimiento. Permite obtener sólo los expedientes de un procedimiento concreto</i>
<i>TpoDate fechaAltaIni</i>	<i>Fecha de alta inicial Permite obtener sólo aquellos expedientes que fueron dados de alta después de la fecha indicada</i>
<i>TpoDate fechaAltaFin</i>	<i>Fecha de alta fin Permite obtener sólo aquellos expedientes que fueron dados de alta antes de la fecha indicada</i>
<i>String soloVigentes</i>	<i>Permite obtener sólo los expedientes que son de un tipo de expediente vigente en el motor. Por defecto 'N' → "S" – Sí → "N" – No</i>

<i>String ordenarPor</i>	<p>Permite ordenar el resultado. Por defecto 'T'. → "T" – Sistema, Tipo de expediente, Procedimiento, Metafase, Fase, Año, Expediente. → "E" – Sistema, Metafase, Fase, Tipo de expediente, Procedimiento, Año, Expediente. → "TEX" – Sistema, Tipo de expediente, Metafase, Fase, Año, Expediente. → "TEV" – Sistema, Procedimiento, Metafase, Fase, Año, Expediente. → "EEX" – Sistema, Metafase, Fase, Tipo de expediente, Año, Expediente. → "EEV" – Sistema, Metafase, Fase, Procedimiento, Año, Expediente. → "TO" – Sistema, Tipo de expediente, Procedimiento, Orden metafase, Metafase, Orden fase, Fase, Año, Expediente. → "EO" – Sistema, Orden metafase, Metafase, Orden fase, Fase, Tipo de expediente, Procedimiento, Año, Expediente.</p>
<i>String pendientes</i>	<p>Permite obtener sólo los expedientes sobre los que el usuario tiene algún permiso para realizar tareas o cambios de fase. Por defecto 'N'. → "S" – Sí → "N" – No</p>
<i>String caducados</i>	<p>Permite obtener sólo los expedientes que se encuentran caducados. Por defecto 'N'. → "S" – Sí → "N" – No</p>
<i>String reservados</i>	<p>Permite obtener sólo los expedientes que se encuentran reservados por el usuario. Por defecto 'N'. → "S" – Sí → "N" – No</p>

Salida

TrExplorador [] *Array de objetos TrExplorador.*

2.4.1.4.4.18 Insertar organismo del expediente

<p>TpoPK insertarOrganismoExpediente(TrOrganismoExpediente orgExp) throws TrException;</p>
--

Inserta una nueva asociación entre un organismo y un expediente.

Entradas

TrOrganismoExpediente orgExp *Organismo del expediente*

Salidas

TpoPK *Identificador de la nueva asociación entre el organismo y el expediente.*

2.4.1.4.4.19 Insertar pago al expediente

TpoPK insertarPagoTasa(TrPagoTasa pagoTasa) throws TrException;

Inserta un pago de tasa a un expediente con un interesado.

Entradas

TrPagoTasa pagoTasa *Pago del expediente*

Salidas

TpoPK *Identificador de la nueva asociación entre el pago y el expediente.*

2.4.1.4.4.20 Creación de relaciones entre expedientes

TpoPK insertarRelacionExpediente (
TpoPK idExpediente,
TrRelacionExpediente relacionExp) throws TrException

Permite crear una nueva relación entre expedientes. Si se indica el REFCOMPONENTE la relación se entiende que es con un expediente externo, siempre que el componente no sea el definido en la constante MOTOR_TRAMITA del sistema por defecto.

Al insertar una nueva relación entre expedientes se toma como referencia el expediente indicado en el parámetro idExpediente. Por ejemplo si queremos insertar una nueva relación entre el expediente 179532 y el 171155 dónde el 179532 sea el “padre” se deben indicar los siguientes atributos:

```
TrRelacionExpediente relacionExp = new TrRelacionExpediente();
relacionExp.setREFEXPEDIENTE(new TpoPK(171155));
relacionExp.setTIPORELACION("M");
apiUI.insertarRelacionExpediente(new TpoPK(179532), relacionExp);
```

La forma correcta de interpretar estos datos será: el “idExpediente” es un expediente “TIPORELACION” del expediente “REFEXPEDIENTE”. Por ejemplo, el expediente 179532 es un expediente “M” (Maestro) del expediente 171155.

Entradas

TpoPK IdExpediente *Identificador del expediente.*
Si el identificador es 'null' se generará una excepción.

TrRelacionExpediente relacionExp *Datos de la relación entre expedientes*

Salida

TpoPK *Identificador de la nueva relación entre expedientes*

2.4.1.4.4.21 Creación de usuarios asignados a un expediente

**void insertarUsuarioAsignadoExpediente (TrUsuarioAsignado usuarioAsig) throws
TrException**

Permite asignar un usuario a un expediente. Para ello se comprueba que el expediente existe y no esté archivado, además de que exista el usuario.

Los campos obligatorios para el usuario asignado son CODUSUARIO, REFEXPEDIENTE, TIPO y FECHAALTA. El identificador del usuario asignado se devuelve en el atributo REFUSUASIGNADO.

Entradas

TrUsuarioAsignado usuarioAsig *Usuario asignado a insertar*

void insertarUsuarioAsignadoExpediente (
TpoPK idExpediente,
TrUsuarioAsignado usuarioAsig) throws TrException
Deprecado

Entradas

TpoPK idExpediente *Identificador del expediente*

TrUsuarioAsignado usuarioAsig *Usuario asignado a insertar*

2.4.1.4.4.22(w) (b) Modificación de expediente

void modificarDatosExpediente (
TpoPK idExpediente,
String numExp,
String tituloExp,
String observaciones,
TpoPK idUniOrg,
TpoPK idUniOrgEnvia,
String urlWanda) throws TrException

Permite modificar los datos de un expediente. Si el identificador del expediente es 'null' o no existe se generará una excepción. Además si el expediente está archivado no se permitirá modificar sus datos, generando una excepción.

A este método se le deben pasar todos los parámetros aunque sólo se quiera modificar uno de ellos.

Si se define un componente w@rdA en el sistema establecido en el TrAPIUI y el documento del expediente está guardado en el mismo w@rdA se actualizan sus datos.

Además en la nueva versión permite indicar la url del expediente en w@ndA.

Si el sistema establecido tiene conexión al bus se informará de los cambios en los datos del expediente. Si no se había informado del expediente previamente, se informará en este momento junto a sus fases e interesados.

Entradas

<i>TpoPK IdExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null' se generará una excepción.</i>
<i>String numExp</i>	<i>Número del expediente para la aplicación cliente</i>
<i>String tituloExp</i>	<i>Título del expediente para la aplicación cliente</i>
<i>String observaciones</i>	<i>Observaciones para el expediente</i>
<i>TpoPK IdUniOrg</i>	<i>Identificador de la unidad orgánica u organismo al que pertenece el expediente</i>
<i>TpoPK idUniOrgEnvia</i>	<i>Identificador de la unidad orgánica u organismo que envía el expediente.</i>
<i>String urlWanda</i>	<i>Url del expediente en w@ndA</i>

void modificarDatosExpediente (

*TpoPK idExpediente,
String numExp,
String tituloExp,
String observaciones,
TpoPK idUniOrg,
TpoPK idUniOrgEnvia) throws TrException*

Entradas

<i>TpoPK IdExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null' se generará una excepción.</i>
<i>String numExp</i>	<i>Número del expediente para la aplicación cliente</i>
<i>String tituloExp</i>	<i>Título del expediente para la aplicación cliente</i>
<i>String observaciones</i>	<i>Observaciones para el expediente</i>
<i>TpoPK IdUniOrg</i>	<i>Identificador de la unidad orgánica u organismo al que pertenece el expediente</i>
<i>TpoPK idUniOrgEnvia</i>	<i>Identificador de la unidad orgánica u organismo que envía el expediente.</i>

2.4.1.4.4.23(b) Modificación del procedimiento seguido

```
void modificarDefProcedimientoExpediente(
    TpoPK idExpediente,
    TpoPK idTipoExp,
    TpoPK idDefProc,
    TpoDate fecha,
    String observaciones) throws TrException
```

Permite modificar, durante la tramitación de un expediente, el procedimiento que está siguiendo, es decir, permite cambiar el tipo de expediente y en consecuencia el procedimiento según la versión. También permite indicar el motivo del cambio de procedimiento para que así conste en la "historia de procedimientos" del expediente mediante el parámetro "observaciones".

Si el sistema establecido tiene conexión al bus se informará la modificación del procedimiento del expediente si el expediente estaba ya informado, si no estaba informado se informarán todos sus datos.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si no se indica identificador del expediente se generará una excepción.
<i>TpoPK idTipoExp</i>	<i>Identificador del tipo de expediente.</i> Si no se indica identificador del tipo de expediente se generará una excepción.
<i>TpoPK idDefProc</i>	<i>Identificador del procedimiento a seguir.</i> Si no se indica identificador del procedimiento se generará una excepción.
<i>TpoDate fecha</i>	<i>Fecha de modificación.</i> Entrada/Salida. Si no se indica fecha de modificación, por defecto se tomará la del sistema.
<i>String observaciones</i>	<i>Observaciones para anotar el motivo de cambio en la "historia de flujos".</i>

2.4.1.4.4.24 Modificación de usuarios asignados a un expediente

```
void modificarUsuarioAsignadoExpediente (TrUsuarioAsignado usuarioAsig)
throws TrException
```

Permite modificar un usuario asignado a un expediente. Para ello se comprueba que el expediente existe y no esté archivado, además de que exista el usuario y que estuviese asignado al expediente.

Entradas

TrUsuarioAsignado usuarioAsig *Usuario asignado a modificar*

```
void modificarUsuarioAsignadoExpediente (
    TpoPK idExpediente,
    TrUsuarioAsignado usuarioAsig) throws TrException
```

Deprecado

Entradas

TpoPK idExpediente *Identificador del expediente*
TrUsuarioAsignado usuarioAsig *Usuario asignado a modificar*

2.4.1.4.4.25 Modificar organismo del expediente

```
void modificarOrganismoExpediente( TrOrganismoExpediente orgExp) throws
TrException;
```

Permite modificar la asociación entre un organismo y un expediente.

Entradas

TrOrganismoExpediente orgExp *Organismo del expediente*

2.4.1.4.4.26 Modificar pago del expediente

```
void modificarPagoTasa( TrPagoTasa pagoTasa) throws TrException;
```

Permite modificar la asociación entre un pago y un expediente.

Entradas

TrPagoTasa pagoTasa *Pago del expediente*

2.4.1.4.4.27 Modificación del propietario del expediente

```
void modificarPropietarioExpediente (
    TpoPK idExpediente,
    String codUsuario) throws TrException
```

Permite modificar el usuario propietario del expediente, siempre que el expediente no esté archivado.

Entradas

<i>TpoPK IdExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null' o el expediente no existe se generará una excepción.
<i>String codUsuario</i>	<i>Código del usuario propietario del expediente</i> Si el identificador es 'null' o el usuario no existe se generará una excepción.

2.4.1.4.4.28 Modificación de relaciones entre expedientes

```
void modificarRelacionExpediente (
    TpoPK idExpediente,
    TrRelacionExpediente relacionExp) throws TrException
```

Permite modificar una relación entre expedientes. Si se indica el REFCOMPONENTE la relación se entiende que es con un expediente externo, siempre que el componente no sea el definido en la constante MOTOR_TRAMITA del sistema por defecto.

En el objeto relacionExp se deben pasar todos los datos, no solo los que se quieran modificar ya que el api modifica todos y pondría a 'null' los que no se rellenen.

Entradas

<i>TpoPK IdExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null' se generará una excepción.
<i>TrRelacionExpediente relacionExp</i>	<i>Datos de la relación entre expedientes a modificar</i>

2.4.1.4.4.29 Datos tipo de pagos

TrDatoTipoPago [] obtenerDatoTipoPago (
 TpoPK idDatoTipoPago,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos de los tipos de pago que se indique en el parámetro *idDatoTipoPago*. Si se envía 'null' se devuelven todos.

Entradas

<i>TpoPK idDatoTipoPago</i>	Identificador del dato del tipo de pago Si el identificador es 'null' devolverá todos
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrDatoTipoPago.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrDatoTipoPago.

Salida

<i>TrDatoTipoPago []</i>	Array de objetos <i>TrDatoTipoPago</i>
---------------------------	--

2.4.1.4.4.30 Obtener número de expedientes

int obtenerExpedientesCount(
 TpoPK idExpediente,
 ClausulaWhere where,
 ClausulaOrderBy orderBy,
 ClausulaExclusion exclusiones) throws TrException;

Devuelve el número de expedientes

Entradas

<i>TpoPK idExpediente</i>	Identificador del expediente. Si el identificador es 'null' se obtienen todos los expedientes,
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrExpediente.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrExpediente.
<i>ClausulaExclusion exclusiones</i>	Entidades a excluir de la consulta. Consultar exclusiones posibles para TrExpediente

Salida

<i>Int</i>	Número de registros
------------	---------------------

2.4.1.4.4.31 Obtención de expedientes reservados por un usuario

<p>TrExpediente[] obtenerExpedientesReservados(TpoPK idStma) throws TrException</p>

Devuelve para un sistema dado, el conjunto de expedientes que el usuario tiene reservados, ya sea el expediente completo o alguna fase del expediente.

Entradas

TpoPK idStma

Identificador del sistema.

Si el identificador es 'null' se generará una excepción.

Salida

TrExpediente []

Array de objetos TrExpediente.

Este método devuelve sólo inicializado el atributo *REFEXP*.

2.4.1.4.4.32 Obtención de expedientes que se encuentran en una fase

<p>TrExpediente[] obtenerExpedientesEnFase(TpoPK idStma, TpoPK idFase) throws TrException</p>

Permite al usuario obtener una lista de expedientes que se encuentran en una fase determinada y para un sistema dado.

Entradas

TpoPK idStma

Identificador del sistema.

Si el identificador es 'null' se generará una excepción.

TpoPK idFase

Identificador de la fase.

Si el identificador es 'null' se generará una excepción.

Salida

TrExpediente []

Array de objetos TrExpediente.

Este método devuelve sólo inicializado el atributo *REFEXP*.

2.4.1.4.4.33 Obtención de expedientes pendientes para un usuario

<p>TrExpediente[] obtenerExpedientesPendientes(TpoPK idSistema) throws TrException</p>
--

Devuelve para un sistema dado, el conjunto de expedientes sobre los que el usuario tiene algún permiso para realizar tareas o cambios de fase (tramitar, generar algún documento, etc).

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema.</i> Si el identificador es 'null' se generará una excepción.
------------------------	---

Salida

<i>TrExpediente []</i>	<i>Array de objetos TrExpediente.</i> Este método devuelve sólo inicializado el atributo REFEXP.
-------------------------	---

2.4.1.4.4.34 Obtener expedientes de la relación interesados

<p>TrExpediente[] obtenerExpsRelacionInteresado(TpoPK idRepresentante, TpoPK idRepresentado) throws TrException</p>
--

Devuelve un conjunto expedientes en los cuales existe la relación entre un representante y un representado.

Entradas

<i>TpoPK idRepresentante</i>	<i>Identificador del interesado representante</i>
<i>TpoPK idRepresentado</i>	<i>Identificador del interesado representado</i>

Salida

<i>TrExpediente []</i>	<i>Array de objetos TrExpediente.</i> Este método devuelve sólo inicializado el atributo REFEXP.
-------------------------	---

2.4.1.4.4.35 Obtener organismos del expediente

```

TrOrganismoExpediente[] obtenerOrganismosExpediente( TpoPK idOrgExp
boolean incluirOrgExp,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws
TrException;

```

Inserta una nueva asociación entre un organismo y un expediente.

Entradas

TpoPK idOrgExp Identificador de la asociación entre el organismo y el expediente.

boolean incluirOrgExp Indica si se incluyen en el conjunto los organismos pertenece, competente, tramita y resuelve que ya existían en la entidad de procedimientos. Si se incluyen, en TEXTOASOCIA se indica el organismo pertenece mediante la cadena PERTENECE, el organismo competente mediante la cadena COMPETENTE, el organismo resuelve mediante la cadena RESUELVE y el organismo tramita mediante la cadena TRAMITA. Para estos organismos del procedimiento, los demás atributos como REFORGPROC, OBSERVACIONES, y fechas de vigencia irán nulas.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrOrganismoExpediente.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrOrganismoExpediente.

Salida

TrOrganismoExpediente [] Array de objetos TrOrganismoExpediente.

2.4.1.4.4.36 Obtener pago del expediente

```

TrPagoTasa[] obtenerPagoTasa( TpoPK idPagoTasa,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws
TrException;

```

Obtienen los registros de pagos de asociados un expediente.

Entradas

TpoPK idPagoTasa Identificador de la asociación entre el pago y el expediente.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrPagoTasa.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrPagoTasa.

Salida

TrPagoTasa []

Array de objetos TrPagoTasa.

2.4.1.4.4.37 Obtener tipos de documentos ENI

TrTipoDocumentoEni[] obtenerTiposDocumentosENI (TpoPK idTipDocEni, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Obtiene un conjunto de tipos de documentos ENI.

Entrada/Salida

TpoPK idTipDocEni *Identificador de tipo de documentos ENI*

ClausulaWhere where *Cláusula where para la consulta*

ClausulaOrderBy orderBy *Cláusula orderBy para la consulta*

Salida

TrTipoDocumentoEni[] *Array de tipos de documentos ENI*

2.4.1.4.5 APIs sobre procedimientos definidos

2.4.1.4.5.1 Obtención de procedimientos definidos

<p>TrDefProcedimiento[] obtenerDefProcedimientosDefinidos(TpoPK idSistema, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Devuelve para un sistema dado, los datos de todos los procedimientos definidos en dicho sistema. Si no se indica sistema, devuelve todos los procedimientos definidos en TREW@.

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema</i> Si el identificador es 'null', el método devolverá todos los procedimientos definidos existentes.
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrDefProcedimiento. Los filtros que usen <i>TrDefProcedimiento.CAMPO_REFSTMA</i> no tendrán efecto si el método recibe un identificador del sistema distinto de 'null'.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrDefProcedimiento.

Salida

TrDefProcedimiento []	Array de objetos TrDefProcedimiento.
------------------------	--------------------------------------

2.4.1.4.5.2 Obtener el número de procedimientos definidos

```
int obtenerDefProcedimientosDefinidosCount(
    TpoPK idSistema,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException
```

Devuelve el número de procedimientos definidos.

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema</i> Si el identificador es 'null', el método devolverá todos los procedimientos definidos existentes.
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrDefProcedimiento. Los filtros que usen <i>TrDefProcedimiento.CAMPO_REFSTMA</i> no tendrán efecto si el método recibe un identificador del sistema distinto de 'null'.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrDefProcedimiento.

Salida

Int	Número de registros
-----	---------------------

2.4.1.4.5.3 Obtención de indicaciones de fichas del procedimiento



TrIndicacionFicha[] obtenerIndicacionesFichaProcedimiento(TpoPK idDefProcedimiento, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Devuelve para un procedimiento dado un conjunto de indicaciones de fichas del procedimiento. Si no se indica procedimiento se devuelven la de todos.

Entradas

TpoPK idDefProcedimiento *Identificador del procedimiento*
Si el identificador es 'null', el método devolverá todas las indicaciones de ficha de todos los procedimientos. Si se especifica un procedimiento, se comprueba que sea correcto, si no lo es se lanzará una excepción.

ClausulaWhere where *Filtro a aplicar.*
Consultar campos posibles para TrIndicacioFicha.

Los filtros que usen TrIndicacioFicha.CAMPO_REFDEFPROC no tendrán efecto si el método recibe un identificador del procedimiento distinto de 'null'.

ClausulaOrderBy orderBy *Ordenación a aplicar.*
Consultar campos posibles para TrIndicacioFicha.

Salida

TrIndicacioFicha [] Array de objetos TrIndicacioFicha.

2.4.1.4.5.4 Obtención de metafases del procedimiento

TrMetafase[] obtenerMetafasesProcedimiento(TpoPK idDefProcedimiento ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Devuelve el conjunto de metafases del procedimiento indicado.

Entradas

TpoPK idDefProcedimiento *Identificador del procedimiento.* Si no se indica o no es válido se generará una excepción.

ClausulaWhere where *Filtro a aplicar.*
Consultar campos posibles para TrMetafase.

ClausulaOrderBy orderBy *Ordenación a aplicar.*
Consultar campos posibles para TrMetafase.

Salida

TrMetafase [] Array de objetos TrMetafase.

2.4.1.4.5.5 Obtención de normativas del procedimiento

TrNormativaProcedimiento[] obtenerNormativasProcedimiento(TpoPK idDefProc, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Devuelve el conjunto de normativas del procedimiento indicado. Si no se indica procedimiento se devuelven todas.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador del procedimiento.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrNormativaProcedimiento.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrNormativaProcedimiento.

Salida

TrNormativaProcedimiento [] Array de objetos TrNormativaProcedimiento.

2.4.1.4.5.6 Obtención de organismos del procedimiento

TrOrganismoProcedimiento[] obtenerOrganismosProcedimiento(TpoPK idOrgProc boolean incluirOrgProc ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Devuelve una lista de organismos asociados al procedimiento.

Entradas

<i>TpoPK idOrgProc</i>	<i>Identificador de la relación entre el organismo y el procedimiento</i>
<i>boolean incluirOrgProc</i>	<i>Indica si se incluyen en el conjunto los organismos pertenece, competente, tramita y resuelve que ya existían en la entidad de procedimientos.</i> Si se incluyen, en TEXTOASOCIA se indica el organismo pertenece mediante la cadena PERTENECE, el organismo competente mediante la cadena COMPETENTE, el organismo resuelve mediante la cadena RESUELVE y el organismo tramita mediante la cadena TRAMITA. Para estos organismos del procedimiento, los demás atributos como REFORGPROC, OBSERVACIONES, y fechas de vigencia irán nulas.
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrOrganismoProcedimiento.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrOrganismoProcedimiento.

Salida

TrOrganismoProcedimiento [] Array de objetos TrOrganismoProcedimiento.

2.4.1.4.5.7 Obtener otros datos

<p>String obtenerOtrosDatos(TpoPK id, String tipo) throws TrException</p>

Permite obtener otros datos de los interesados, expedientes, documentos o procedimientos según se indique en el parámetro *tipo*.

Entradas

<i>TpoPK id</i>	Identificador del interesado, expediente o procedimiento. Si el identificador es 'null' o no válido se generará una excepción.
<i>String tipo</i>	Indica el tipo del que se quieren obtener otros datos. → "P" – Procedimiento → "I" – Interesado → "E" – Expediente → "D" – Documento → "ENI" – Expediente electrónico → "ENIDOC" – Documento electrónico Si el tipo es 'null' o no válido se generará una excepción.

Salida

<i>String</i>	Otros datos del interesado, expediente o procedimiento.
---------------	---

2.4.1.4.5.8 Obtención de plantillas del procedimiento

<p>TrPlantillaProcedimiento[] obtenerPlantillasProcedimiento(TpoPK idDefProc, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>

Devuelve el conjunto de plantillas del procedimiento indicado. Si no se indica procedimiento se devuelven todas.

Entradas

<i>TpoPK idDefProc</i>	Identificador del procedimiento.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrPlantillaProcedimiento.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrPlantillaProcedimiento.

Salida

TrPlantillaProcedimiento [] Array de objetos TrPlantillaProcedimiento.

2.4.1.4.5.9 Obtención de procedimientos para el usuario

TrDefProcedimiento[] obtenerProcedimientosUsuario (
String codUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve una lista de procedimientos en los que el usuario indicado tiene algún perfil de usuario. Si el codUsuario que se indica es *null* o vacío se realiza la consulta para el usuario establecido en el TrAPIUI.

Entradas

<i>String codUsuario</i>	<i>Identificador del usuario</i> Si el identificador es 'null' se usa el código del usuario establecido en el TrAPIUI.
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrDefProcedimiento.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrDefProcedimiento.

Salida

TrDefProcedimiento []	Array de objetos TrDefProcedimiento.
------------------------	--------------------------------------

2.4.1.4.5.10 Obtener procedimiento padre de un reutilizable

TpoPK obtenerProcPadreDeReutilizable(
TpoPK refExpFase) throws TrException

Obtiene la referencia del procedimiento o del procedimiento principal si la referencia en fase pertenece a un reutilizable.

Entradas

<i>TpoPK refExpFase</i>	<i>Identificador del expediente en fase</i>
-------------------------	---

Salida

TpoPK	Identificador del procedimiento
-------	---------------------------------

2.4.1.4.5.11 Obtención de reutilizables del procedimiento

TpoPK [] obtenerReutilizablesProcedimiento (TpoPK idDefProc, TpoPK idSistema) throws TrException

Devuelve los identificadores de los procedimientos que actúan como módulos reutilizables dentro de un procedimiento.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador del procedimiento.</i>
<i>TpoPK idSistema</i>	<i>Identificador del sistema.</i>

Salida

TpoPK [] Array de identificadores de procedimientos.

2.4.1.4.5.12 Conjunto de fases y transiciones posibles en un procedimiento

TrTransicionDefProcedimiento[] obtenerTransicionesDefProcedimiento(TpoPK idDefProc, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite a un usuario obtener para un procedimiento dado, el conjunto de transiciones posibles que se pueden realizar y las fases en las que derivan sin tener en cuenta el perfil del usuario.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrTransicionDefProcedimiento.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrTransicionDefProcedimiento.

Salida

TrTransicionDefProcedimiento [] *Array de objetos TrTransicionDefProcedimiento.*

2.4.1.4.6 APIs sobre tramitación (transiciones y fases)

2.4.1.4.6.1(b) Deshacer el último paso de un expediente

```
TrMensajeCondicionAccion[ ] deshacerTramitacion(  
TpoPK idExpXFase ) throws TrException
```

Permite al usuario deshacer el paso dado en la tramitación de un expediente siempre que dicho paso corresponda a una situación actual del expediente, y sólo si el usuario tiene los permisos suficientes y además la transición no es un evento de salida, tipo 'ES' (abandona la fase actual). Devuelve un conjunto de mensajes o avisos derivados de las condiciones y acciones al deshacer la transición:

Si el sistema establecido tiene conexión al bus se informará del cambio de fases del expediente, además de los interesados en el expediente. Si el expediente no había sido informado previamente se informará con todos sus datos en este momento.

Entradas

TpoPK idExpXFase

Identificador del expediente en fase

Si el identificador es 'null', o no válido, se generará una excepción.

Salida

TrMensajeCondicionAccion [] Array de objetos TrMensajeCondicionAccion.

2.4.1.4.6.2 Eliminar la reserva de un expediente o de alguna de sus fases

```
void eliminarReservaExpediente(  
TpoPK idExpediente,  
TpoPK idFase) throws TrException
```

Método complementario del anterior. Permite eliminar las reservas sobre un expediente de un sistema dado. Si se indica la fase el desbloqueo sólo es a nivel de la fase.

Los usuarios "administradores" pueden desbloquear aunque existan bloqueos de otros usuarios.

Entradas

TpoPK idExpediente

Identificador del expediente.

Si el identificador es 'null', o no válido, se generará una excepción.

TpoPK idFase

Identificador de la fase.

Opcional

2.4.1.4.6.3(b) Enviar expediente

```
void enviarExpedienteA(
    TpoPK idExpediente,
    TrEnviarA[ ] datosTransicion,
    String bloqueo,
    String usuario,
    TrFaseActualExpediente[ ] faseActualExp ) throws TrException
```

Permite enviar el expediente a una situación concreta. Se debe indicar los datos de la transición que se desea realizar mediante un array de objetos TrEnviarA que se pasa como parámetro.

Si el sistema establecido tiene conexión al bus se informará del cambio de fases del expediente, además de los interesados en el expediente. Si el expediente no había sido informado previamente se informará con todos sus datos en este momento.

Entradas

<i>TpoPK</i> idExpediente	<i>Identificador del expediente.</i>
<i>TrEnviarA</i> [] datosTransicion	<i>Array de objetos TrEnviarA.</i>
<i>String</i> bloqueo	Indica si se permite la reserva del expediente o de la fase. → “ E ” – Permite la reserva del expediente → “ F ” – Permite la reserva de la fase
<i>String</i> usuario	Indica el usuario que reserva.
<i>TrFaseActualExpediente</i> [] faseActualExp	Array con fases del expediente desde las cuales se quiere hacer “enviar a ...” Es necesario inicializar correctamente los atributos <i>FASE</i> , <i>REFFASE</i> Y <i>REFFASEPADRE</i> en TrFaseActualExpediente. El resto de atributos son ignorados.

A través del parámetro *faseActualExp* se indica la fase del expediente desde la cual se quiere hacer “enviar a...”, de esta forma se puede anotar la fecha de salida desde esta/s fase/s (ya que el expediente se puede encontrar en varias a la vez). Si no se indica nada la nueva situación es una más para el expediente, es decir, el expediente queda en la situación que estuviera más la nueva situación a la que enviamos.

2.4.1.4.6.4 Evaluar condiciones de una transición

```

TrMensajeCondicionAccion[ ] evaluarCondicionesTransicion(
    TpoPK idExpediente,
    TpoPK idTransicion,
    TpoPK idDefProc,
    String comprobar,
    TpoBoolean resultado,
    TpoDate fecha ) throws TrException
    
```

Permite al usuario evaluar el conjunto de condiciones asociadas a una transición para un expediente dado en un procedimiento también dado.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Opcional.</i>
<i>TpoPK idTransicion</i>	<i>Identificador de la transición. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idExpFase</i>	<i>Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.</i>
<i>String comprobar</i>	Usado para indicar cuáles son las condiciones que se quieren evaluar: → "T" – Evaluar las condiciones definidas para tramitar. → "D" – Evaluar las condiciones definidas para deshacer. → "A" – Evaluar ambos tipos de condiciones. → "V" – Evaluar las condiciones definidas para visualización.
<i>TpoBoolean resultado</i>	Indica si alguna de las condiciones es obligatoria y no se cumple. → 'false' – Alguna da las condiciones es obligatoria y no se cumple. → 'true' – No existen condiciones obligatorias que no se cumplan.
<i>TpoDate fecha</i>	Entrada/Salida. Fecha Si no se indica fecha, tomará por defecto la del sistema.

Salida

TrMensajeCondicionAccion [] Array de objetos TrMensajeCondicionAccion con los mensajes resultado de evaluar las condiciones.

```

TrMensajeCondicionAccion[ ] evaluarCondicionesTransicion(
    TpoPK idExpediente,
    TpoPK idTransicion,
    TpoPK idDefProc,
    TpoPK idExpFase,
    String comprobar,
    TpoBoolean resultado,
    TpoDate fecha ) throws TrException
    
```

Entradas

<i>TpoPK idExpediente</i>	Identificador del expediente. <i>Opcional.</i>
<i>TpoPK idTransicion</i>	Identificador de la transición. Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.
<i>String comprobar</i>	Usado para indicar cuáles son las condiciones que se quieren evaluar: → "T" – Evaluar las condiciones definidas para tramitar. → "D" – Evaluar las condiciones definidas para deshacer. → "A" – Evaluar ambos tipos de condiciones. → "V" – Evaluar las condiciones definidas para visualización.
<i>TpoBoolean resultado</i>	Indica si alguna de las condiciones es obligatoria y no se cumple. → 'false' – Alguna da las condiciones es obligatoria y no se cumple. → 'true' – No existen condiciones obligatorias que no se cumplan.
<i>TpoDate fecha</i>	Entrada/Salida. Fecha Si no se indica fecha, tomará por defecto la del sistema.

Salida

TrMensajeCondicionAccion [] Array de objetos TrMensajeCondicionAccion con los mensajes resultado de evaluar las condiciones.

2.4.1.4.6.5(b) Modificar los datos de una fase del expediente

```
void modificarDatosFaseActual(  
TrFaseActualExpediente datoEvolucion ) throws TrException
```

Permite modificar los datos de un paso concreto en la evolución de un expediente. Sólo se permite modificar los datos de aquellas situaciones en las que se encuentre el expediente, que hayan sido realizadas por el usuario que intenta modificarlos y siempre que no estén reservadas por otro usuario, a menos de que el atributo *comprobarBloqueo* sea false, en ese caso puede realizar la modificación cualquier usuario.

Si el sistema establecido tiene conexión al bus se informará del cambio de fases del expediente. Si el expediente no había sido informado previamente se informará con todos sus datos en este momento.

Entradas

TrFaseActualExpediente datoEvolucion *Situación del expediente, a modificar.*
Se debe aportar los atributos FECHAENTRADA / FECHALIMITE / OBSERVACIONES / REFEXPFAS para la modificación aunque sólo uno de ellos fuese el cambiara.

Nota: La actualización se hace a nivel de todo el registro, por lo que no sólo se indican los campos que cambian sino también los que no cambian. Es decir, aunque por ejemplo, sólo queremos modificar la fecha de salida debemos indicar los demás datos que componen el registro, si ponemos null TREW@ intentará poner nulos siempre que sea posible.

2.4.1.4.6.6 Acciones asociadas a una transición

TrAccionTransicion[] obtenerAccionesTransicion(
TpoPK idDefProc,
TpoPK idTransicion,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite al usuario obtener los datos del conjunto de acciones (válidas) asociadas a una transición y en un procedimiento dados.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idTransicion</i>	<i>Identificador de la transición. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrAccionTransicion.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrAccionTransicion.</i>

Salida

TrAccionTransicion []	<i>Array de objetos TrAccionTransicion.</i>
------------------------	---

2.4.1.4.6.7 Condiciones, acciones y avisos

TrCondicionAccionAviso[] obtenerCondicionesAccionesAvisos(
TpoPK idDefProc,
TpoPK idTranDocuOtTa,
String tranDocuOtTa,
String tipo,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve todas las condiciones, acciones y avisos del procedimiento para el tipo indicado en el parámetro tranDocuOtTa. El valor de este parámetro puede ser "T" para las condiciones, acciones y avisos al tramitar, "D" de los documentos y "O" de otras tareas. Además permite filtrar por el parámetro tipo que puede ser "C" condición, "A" acción y "W" aviso.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idTranDocuOtTa</i>	<i>Identificador de la transición, documento u otra tarea.</i>
<i>String tranDocuOtTa</i>	<i>Indica qué condiciones, acciones y avisos se quieren obtener. → 'T' – Condiciones, acciones y avisos de la transición → 'D' – Condiciones, acciones y avisos del documento → 'O' – Condiciones, acciones y avisos de otras tareas</i>
<i>String tipo</i>	<i>Indica el tipo de datos que se quieren obtener → 'C' –Condiciones → 'A' – Acciones → 'W' – Avisos Si se le pasa null se obtienen todos.</i>

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTransicion.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTransicion.

Salida

TrCondicionAccionAviso[] Array de objetos *TrCondicionAccionAviso*.

2.4.1.4.6.8 Condiciones asociadas a una transición

TrCondicionTransicion[] obtenerCondicionesTransicion(
 TpoPK idDefProc,
 TpoPK idTransicion,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite al usuario obtener los datos del conjunto de condiciones (válidas) asociadas a una transición en un procedimiento dados.

Entradas

TpoPK idDefProc Identificador de la definición del procedimiento.
Si el identificador es 'null', o no válido, se generará una excepción.

TpoPK idTransicion Identificador de la transición.
Si el identificador es 'null', o no válido, se generará una excepción.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrCondicionTransicion.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrCondicionTransicion.

Salida

TrCondicionTransicion [] Array de objetos *TrCondicionTransicion*.

2.4.1.4.6.9 Datos de una transición

TrTransicion[] obtenerDatosTransicion(
 TpoPK idTransicion,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Devuelve los datos de una transición pasada como parámetro. Si la transición no existiese se generaría una excepción.

Entradas

TpoPK idTransicion Identificador de la transición
Si el identificador es 'null', o no válido, se generará una excepción.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTransicion.
En este api no se tiene en cuenta el CAMPO_REFTRANSICION ya que se recibe el parámetro idTransición.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTransicion.

Salida

TrTransicion [] Array de objetos *TrTransicion*.

2.4.1.4.6.10 Obtención de eventos posibles

```

TrTransicion[ ] obtenerEventosPosibles(
    TpoPK idDefProc,
    TpoPK idExpediente,
    TpoDate fecha,
    String condVisual,
    boolean soloUsuario,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException

```

Permite obtener los posibles eventos para un expediente a una fecha dada (si no se indica se toma la fecha del sistema). Esto permite definir eventos que dependen de una fecha, de esta forma se permiten definir condiciones con un parámetro fecha.

Si las condiciones definidas no necesitan los parámetros "idExpediente " y "fecha", éstos se pueden pasar nulos.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Opcional</i>
<i>TpoDate fecha</i>	<i>Fecha. Si no se indica fecha, tomará por defecto la del sistema.</i>
<i>String condVisual</i>	<i>Indica si se quieren obtener las transiciones en las que se cumplan las condiciones de visualización o no. → 'S' – Sólo las que cumplan las condiciones (Por defecto) → 'N' – Sólo las que no cumplan las condiciones → 'T' – Todas</i>
<i>boolean soloUsuario</i>	<i>Indica si se filtra por el usuario y su perfil. Por defecto true</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrTransicion.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrTransicion.</i>

Salida

TrTransicion []	Array de objetos TrTransicion que representan los eventos posibles que se pueden realizar.
------------------	--

```

TrTransicion[ ] obtenerEventosPosibles(
    TpoPK idDefProc,
    TpoPK idExpediente,
    TpoDate fecha,
    ClausulaWhere where,
    ClausulaOrderBy orderBy) throws TrException
Deprecado

```

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i>
<i>TpoDate fecha</i>	Opcional <i>Fecha.</i> Si no se indica fecha, tomará por defecto la del sistema.
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrTransicion.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrTransicion.

Salida

TrTransicion []	Array de objetos TrTransicion que representan los eventos posibles que se pueden realizar.
------------------	--

TrTransicion[] obtenerEventosPosibles(
TpoPK idDefProc,
TpoPK idExpediente,
TpoPK idExpFase,
TpoDate fecha,
String condVisual,
boolean soloUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i>
<i>TpoPK idExpFase</i>	Opcional <i>Identificador del expediente en fase</i>
<i>TpoDate fecha</i>	<i>Fecha.</i> Si no se indica fecha, tomará por defecto la del sistema.
<i>String condVisual</i>	<i>Indica si se quieren obtener las transiciones en las que se cumplan las condiciones de visualización o no.</i> → 'S' – Sólo las que cumplan las condiciones (Por defecto) → 'N' – Sólo las que no cumplan las condiciones → 'T' – Todas
<i>boolean soloUsuario</i>	<i>Indica si se filtra por el usuario y su perfil. Por defecto true</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrTransicion.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrTransicion.

Salida

TrTransicion []	Array de objetos TrTransicion que representan los eventos posibles que se pueden realizar.
------------------	--

2.4.1.4.6.11 Evolución en fases de un expediente

TrEvolucionExpediente[] obtenerEvolucionExpediente(

TpoPK idExpediente,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve, para un expediente, los datos acerca de la evolución del expediente dentro del procedimiento, es decir, la historia de situaciones por donde ha pasado el expediente.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrEvolucionExpediente.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrEvolucionExpediente.

Salida

TrEvolucionExpediente []	Array de objetos TrEvolucionExpediente recogiendo la historia de situaciones por las que ha pasado un expediente.
---------------------------	---

2.4.1.4.6.12 Obtener el número de evolución expediente

int obtenerEvolucionExpedienteCount(
TpoPK idExpediente,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve el número de registros de la evolución de expedientes.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrEvolucionExpediente.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrEvolucionExpediente.

Salida

Int	Número de registros
-----	---------------------

2.4.1.4.6.13 Obtención de la(s) fase(s) actual(es) de un expediente

TrFaseActualExpediente[] obtenerFaseActualExpediente(
TpoPK idExpediente,

ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Dado un expediente, devuelve los datos de las fases o situaciones en la que se encuentra actualmente el expediente.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null', el método devolverá todas las versiones definidas en TREW@.
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrFaseActualExpediente.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrFaseActualExpediente.

Salida

TrFaseActualExpediente [] *Array de objetos TrFaseActualExpediente.*

En este método hay que tener en cuenta que los filtros y ordenaciones sobre los datos se hacen por niveles (en caso de módulos reutilizables), es decir, en caso de que un determinado filtro no se cumpla para un nivel, hace que no se devuelva dicho nivel ni los que dependen de él.

2.4.1.4.6.14 Obtener fases del expediente

TrFaseExpediente[] obtenerFasesExpediente(
TpoPK idExpediente,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve para el expediente indicado, si no se indica se devuelven los datos de todos los expedientes, los datos acerca de su evolución dentro del procedimiento, es decir, la historia de fases por las que ha pasado el expediente. La información de las fases bloqueadas se devolverá a cualquier usuario siempre y cuando el atributo *comprobarBloqueo* sea false.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrFaseExpediente.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrFaseExpediente.

Salida

TrFaseExpediente [] *Array de objetos TrFaseExpediente recogiendo la historia de situaciones por las que ha pasado un expediente.*

2.4.1.4.6.15 Fases finales de una transición

TrFase[] obtenerFasesFinTransicion(
TpoPK idTransicion,

TpoPK idDefProc,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener la fase fin en la que termina la transición. En caso de una transición de división devolverá más de una fase final.

Se comprueba que la transición exista, además de que el procedimiento exista y no esté bloqueado, generando una excepción en caso contrario.

Entradas

<i>TpoPK idTransicion</i>	<i>Identificador de la transición</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrTransicion.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrTransicion.

Salida

<i>TrFase []</i>	<i>Array de objetos TrFase.</i>
-------------------	---------------------------------

2.4.1.4.6.16 Fases de un procedimiento

TrFase[] obtenerFasesProcedimiento(
TpoPK idDefProcedimiento,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve para el procedimiento indicado el conjunto de fases que lo componen. Para ello se obtienen las fases cuyas transiciones sean válidas y están asociadas al procedimiento. Se debe indicar el identificador del procedimiento obligatoriamente y se pueden realizar filtros y ordenaciones con los campos disponibles en la clase TrFase.

Entradas

<i>TpoPK idDefProcedimiento</i>	<i>Identificador del procedimiento</i>
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrFase.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrFase.

Salida

<i>TrFase []</i>	<i>Array de objetos TrFase recogiendo todas las fases que existen en el procedimiento indicado.</i>
-------------------	---

2.4.1.4.6.17 Fecha de un acto para un expediente

TpoDate obtenerFechaActoAdmExpediente(
TpoPK idExpediente,

TpoPK idTipoActo) throws TrException

Permite al usuario obtener la fecha en la que se produjo un acto administrativo para un expediente (la más reciente).

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>TpoPK idTipoActo</i>	<i>Identificador del tipo de acto.</i>

Salida

<i>TrDate</i>	<i>Fecha en la que se produjo el acto administrativo.</i>
---------------	---

2.4.1.4.6.18 Obtener transiciones permitidas

TrTransicion[] obtenerTransicionesPermitidas(

TpoPK idFase,
TpoPK idDefProc,
TpoPK idExpediente,
TpoDate fecha,
String condVisual,
boolean soloUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite a un usuario obtener para una fase y un procedimiento dado, el conjunto de transiciones posibles que se pueden realizar según el perfil del usuario. Permite filtrar las transiciones que tengan condiciones de visualización.

Entradas

<i>TpoPK idFase</i>	<i>Identificador de la fase. Si el identificador es 'null', el método devolverá las transiciones de entrada que dan comienzo a la tramitación.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Opcional</i>
<i>TpoDate fecha</i>	<i>Fecha. Si no se indica fecha, tomará por defecto la del sistema.</i>
<i>String condVisual</i>	<i>Indica si se quieren obtener las transiciones en las que se cumplan las condiciones de visualización o no. → 'S' – Sólo las que cumplan las condiciones (Por defecto) → 'N' – Sólo las que no cumplan las condiciones → 'T' – Todas</i>
<i>boolean soloUsuario</i>	<i>Indica si se filtra por el usuario y su perfil. Por defecto true</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrTransicion.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrTransicion.</i>

Salida

<i>TrTransicion []</i>	<i>Array de objetos TrTransicion que representan las transiciones posibles que se pueden realizar.</i>
-------------------------	--

En este método hay que tener en cuenta que los filtros y ordenaciones sobre los datos se hacen por niveles (en caso de módulos reutilizables), es decir, en caso de que un determinado filtro no se cumpla para un nivel, hace que no se devuelva dicho nivel ni los que dependen de él.

<p>TrTransicion[] obtenerTransicionesPermitidas(TpoPK idFase, TpoPK idDefProc, TpoPK idExpediente, TpoDate fecha, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>	<i>Deprecado</i>
---	------------------

Entradas

<i>TpoPK idFase</i>	<i>Identificador de la fase.</i> Si el identificador es 'null', el método devolverá las transiciones de entrada que dan comienzo a la tramitación.
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento.</i> Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i>
<i>TpoDate fecha</i>	Opcional <i>Fecha.</i> <i>Si no se indica fecha, tomará por defecto la del sistema.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrTransicion.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrTransicion.

Salida

TrTransicion []	Array de objetos TrTransicion que representan las transiciones posibles que se pueden realizar.
------------------	---

<p>TrTransicion[] obtenerTransicionesPermitidas(TpoPK idFase, TpoPK idDefProc, TpoPK idExpediente, TpoPK idExpFase, TpoDate fecha, String condVisual, boolean soloUsuario, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>

Entradas

<i>TpoPK idFase</i>	<i>Identificador de la fase.</i> Si el identificador es 'null', el método devolverá las transiciones de entrada que dan comienzo a la tramitación.
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento.</i> Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Opcional

<i>TpoPK idExpFase</i>	<i>Identificador del expediente en fase</i>
<i>TpoDate fecha</i>	<i>Fecha.</i> <i>Si no se indica fecha, tomará por defecto la del sistema.</i>
<i>String condVisual</i>	<i>Indica si se quieren obtener las transiciones en las que se cumplan las condiciones de visualización o no.</i> → 'S' – Sólo las que cumplan las condiciones (Por defecto) → 'N' – Sólo las que no cumplan las condiciones → 'T' – Todas
<i>boolean soloUsuario</i>	<i>Indica si se filtra por el usuario y su perfil. Por defecto true</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrTransicion.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrTransicion.

Salida

TrTransicion []	Array de objetos TrTransicion que representan las transiciones posibles que se pueden realizar.
------------------	---

2.4.1.4.6.19 Reservar un expediente o alguna de sus fases

```
void reservarExpediente(
    TpoPK idExpediente,
    TpoPK idFase,
    TpoPK expFase) throws TrException
```

Permite bloquear un expediente o una fase concreta del expediente. Si no se indica fase, la reserva es al nivel de expediente y todas las situaciones quedan bloqueadas por el usuario. Si la fase

representa un procedimiento reutilizable, un bloqueo de fase implica el bloqueo de todas las situaciones en las que se encuentre el expediente dentro del procedimiento reutilizable.

Los usuarios “administradores” pueden bloquear aunque existan bloqueos de otros usuarios.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idFase</i>	<i>Identificador de la fase.</i>
<i>TpoPK expFase</i>	<i>Opcional</i> Identificador de la instancia de la fase.

<p>void reservarExpediente(TpoPK idExpediente, TpoPK idFase) throws TrException</p>

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idFase</i>	<i>Identificador de la fase.</i>
	<i>Opcional</i>

2.4.1.4.6.20(b) Tramitación de un expediente

<p>TrMensajeCondicionAccion[] tramitarExpediente(TpoPK idExpediente, TpoPK idFase, TpoPK idTransicion, TpoPK idDefProc, TpoDate fecha, TpoDate fechaLimite, String bloqueo,</p>

String observaciones,
boolean comprobarTerminadas) throws TrException

Método que realiza en un procedimiento determinado para el expediente que se indica como parámetro, la transición dada, desde una fase dada, para un expediente también dado.

Si el sistema establecido tiene conexión al bus se informará del cambio de fases del expediente, además de los interesados en el expediente. Si el expediente no había sido informado previamente se informará con todos sus datos en este momento.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i>
<i>TpoPK idFase</i>	<i>Identificador de la fase.</i>
<i>TpoPK idTransicion</i>	<i>Identificador de la transición.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento.</i>
<i>TpoDate fecha</i>	<i>Fecha.</i> Si no se indica fecha, por defecto se tomará la del sistema.
<i>TpoDate fechaLimite</i>	<i>Fecha límite de estancia en la fase.</i>
<i>String bloqueo</i>	<i>Indica si el usuario quiere bloquear la fase ('F') o el expediente ('E').</i>
<i>String observaciones</i>	<i>Observaciones de tramitación a la fase.</i> Esta cadena no puede tener una longitud superior a 250 caracteres.
<i>boolean comprobarTerminadas</i>	<i>Indica si se comprueban que las tareas estén terminadas.</i> Por defecto false.

Salida

TrMensajeCondicionAccion [] Array de objetos *TrMensajeCondicionAccion*.

Devuelve un array de mensajes (avisos, errores) derivados de tramitar el expediente y comprobar sus condiciones o realizar las acciones asociadas a la transición.

Si existen condiciones obligatorias definidas sobre la transición y alguna no se cumple, no se dejará tramitar a no ser que el usuario tenga permisos de administración.

<p>TrMensajeCondicionAccion[] tramitarExpediente(TpoPK idExpediente, TpoPK idFase, TpoPK idTransicion, TpoPK idDefProc, TpoDate fecha, String bloqueo, String observaciones) throws TrException</p>	<i>Deprecado</i>
---	------------------

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i>
<i>TpoPK idFase</i>	<i>Identificador de la fase.</i>
<i>TpoPK idTransicion</i>	<i>Identificador de la transición.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento.</i>
<i>TpoDate fecha</i>	<i>Fecha.</i>
<i>String bloqueo</i>	<i>Si no se indica fecha, por defecto se tomará la del sistema. Indica si el usuario quiere bloquear la fase ('F') o el expediente ('E').</i>
<i>String observaciones</i>	<i>Observaciones de tramitación a la fase. Esta cadena no puede tener una longitud superior a 250 caracteres.</i>

Salida

TrMensajeCondicionAccion [] Array de objetos *TrMensajeCondicionAccion*.

TrMensajeCondicionAccion [] tramitarExpediente(

TpoPK idExpediente,
TpoPK idFase,
TpoPK idTransicion,
TpoPK idDefProc,
TpoDate fecha,
TpoDate fechaLimite,
TpoPK expFase,
String bloqueo,
String observaciones,
boolean comprobarTerminadas) throws TrException

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente.</i>
<i>TpoPK idFase</i>	<i>Identificador de la fase.</i>
<i>TpoPK idTransicion</i>	<i>Identificador de la transición.</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición de procedimiento.</i>
<i>TpoDate fecha</i>	<i>Fecha.</i>
<i>TpoDate fechaLimite</i>	<i>Si no se indica fecha, por defecto se tomará la del sistema. Fecha límite de estancia en la fase.</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase</i>
<i>String bloqueo</i>	<i>Indica si el usuario quiere bloquear la fase ('F') o el expediente ('E').</i>
<i>String observaciones</i>	<i>Observaciones de tramitación a la fase. Esta cadena no puede tener una longitud superior a 250 caracteres.</i>
<i>boolean comprobarTerminadas</i>	<i>Indica si se comprueban que las tareas estén terminadas. Por defecto false.</i>

Salida

TrMensajeCondicionAccion [] Array de objetos *TrMensajeCondicionAccion*.



2.4.1.4.7 APIs sobre bloques de datos (tareas II)

2.4.1.4.7.1 Descartar tarea del expediente

```
void descartarTareaExpediente(  
    TpoPK idTareaExp,  
    String tipoTarea,  
    TpoDate fecha) throws TrException
```

Permite descartar una tarea de un expediente. El tipo de la tarea puede ser: “**I**”: Para un documento incorporado; “**G**”: Para un documento generado; “**M**”: Tarea del tipo manipular datos; “**O**”: Otros tipos de tarea.

Entradas

<i>TpoPK idTareaExp</i>	<i>Identificador de la tarea Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String tipoTarea</i>	<i>Tipo de tarea → "I" – Incorporar → "G" – Generar → "M" – Manipular datos → "O" – Otras</i>
<i>TpoDate fecha</i>	<i>Fecha de descarte de la tarea Si se pasa 'null' por defecto se coge la del sistema</i>

2.4.1.4.7.2 Eliminar consulta de certificado de un interesado en un documento

void eliminarConsultaCertificado(TpoPK idConsulta) throws TrException

Permite eliminar los datos de una consulta de certificado para un interesado en un documento del expediente.

Entradas

<i>TpoPK idConsulta</i>	<i>Identificador de la consulta Si el identificador es 'null', o no válido, se generará una excepción</i>
-------------------------	---

2.4.1.4.7.3 Eliminar tarea del expediente

void eliminarTareaExpediente(TpoPK idTareaExp, String tipoTarea) throws TrException

Permite eliminar una tarea de un expediente. El tipo de la tarea puede ser: "I": Para un documento incorporado; "G": Para un documento generado; "M": Tarea del tipo manipular datos; "O": Otros tipos de tarea.

Entradas

<i>TpoPK idTareaExp</i>	<i>Identificador de la tarea Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String tipoTarea</i>	<i>Tipo de tarea → "I" – Incorporar → "G" – Generar → "M" – Manipular datos → "O" – Otras</i>

2.4.1.4.7.4 Evaluar condiciones de otras tareas

TrMensajeCondicionAccion[] evaluarCondicionesOtrasTareas(TpoPK idExpediente, TpoPK idTareaFase, TpoPK idDefProc, TpoPK idExpFase, String comprobar, TpoBoolean resultado, TpoDate fecha) throws TrException

Permite al usuario evaluar el conjunto de condiciones asociadas a una tarea para un expediente dado en un procedimiento también dado.

Entradas

<i>TpoPK idExpediente</i>	Identificador del expediente. Opcional.
<i>TpoPK idTareaFase</i>	Identificador de la tarea en fase. Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idExpFase</i>	Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.
<i>String comprobar</i>	Usado para indicar cuáles son las condiciones que se quieren evaluar: → "I" – Evaluar las condiciones definidas al iniciar la tarea → "V" – Evaluar las condiciones definidas para visualización.
<i>TpoBoolean resultado</i>	Indica si alguna de las condiciones es obligatoria y no se cumple. → 'false' – Alguna da las condiciones es obligatoria y no se cumple. → 'true' – No existen condiciones obligatorias que no se cumplan.
<i>TpoDate fecha</i>	Entrada/Salida. Fecha Si no se indica fecha, tomará por defecto la del sistema.

Salida

TrMensajeCondicionAccion [] Array de objetos TrMensajeCondicionAccion con los mensajes resultado de evaluar las condiciones.

TrMensajeCondicionAccion[] evaluarCondicionesOtrasTareas(
TpoPK idExpediente,
TpoPK idTareaFase,
TpoPK idDefProc,
String comprobar,
TpoBoolean resultado,
TpoDate fecha) throws TrException

Entradas

<i>TpoPK idExpediente</i>	Identificador del expediente. Opcional.
<i>TpoPK idTareaFase</i>	Identificador de la tarea en fase. Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.
<i>String comprobar</i>	Usado para indicar cuáles son las condiciones que se quieren evaluar: → "I" – Evaluar las condiciones definidas al iniciar la tarea → "V" – Evaluar las condiciones definidas para visualización.

TpoBoolean resultado Indica si alguna de las condiciones es obligatoria y no se cumple.
 → **'false'** – Alguna da las condiciones es obligatoria y no se cumple.
 → **'true'** – No existen condiciones obligatorias que no se cumplan.

TpoDate fecha Entrada/Salida.
 Fecha
 Si no se indica fecha, tomará por defecto la del sistema.

Salida

TrMensajeCondicionAccion [] Array de objetos TrMensajeCondicionAccion con los mensajes resultado de evaluar las condiciones.

2.4.1.4.7.5 Modificar estado de una tarea del expediente

```
void modificarEstadoOtrasTareas(
    TpoPK idTarea,
    String estado,
    TpoDate fecha) throws TrException
```

Permite modificar el estado de la tarea que se indique en el parámetro idTarea. El parámetro estado indica el nuevo estado de la tarea y el parámetro fecha sólo tendrá en cuenta para los estados "D" y "F".

Entradas

TpoPK idTarea Identificador de la tarea
 Si el identificador es 'null', o no válido, se generará una excepción

String estado Estado de la tarea
 → "I" – Iniciada
 → "D" – Descartada
 → "F" – Finalizada

TpoDate fecha Fecha de descarte o finalización de la tarea.
 Si es 'null' se coge la del sistema

2.4.1.4.7.6 Tareas del expediente

```
TrTareaExpediente[ ] obtenerTareasExpediente(
    TpoPK idExpediente,
    boolean soloUsuario,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException
```

Devuelve un conjunto de tareas definidas para uno o varios expedientes. Si se indica el parámetro soloUsuario a **true** devolverá solo aquellas tareas para las que el usuario tenga permisos establecidos.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente Si no se indica se devuelven las tareas de todos los expedientes</i>
<i>boolean soloUsuario</i>	<i>Indica si se filtra por el usuario establecido en el api y su perfil</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrTareaExpediente.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrTareaExpediente.</i>

Salida

<i>TrTareaExpediente []</i>	<i>Array de objetos TrTareaExpediente.</i>
-----------------------------	--

2.4.1.4.7.7 Finalizar tarea del expediente

```
void finalizarTareaExpediente(
    TpoPK idTareaExp,
    String tipoTarea,
    TpoDate fecha) throws TrException
```

Permite finalizar una tarea de un expediente. El tipo de la tarea puede ser: **"I"**: Para un documento incorporado; **"G"**: Para un documento generado; **"M"**: Tarea del tipo manipular datos; **"O"**: Otros tipos de tarea.

Entradas

<i>TpoPK idTareaExp</i>	<i>Identificador de la tarea Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String tipoTarea</i>	<i>Tipo de tarea → "I" – Incorporar → "G" – Generar → "M" – Manipular datos → "O" – Otras</i>
<i>TpoDate fecha</i>	<i>Fecha fin de la tarea Si se pasa 'null' por defecto se coge la del sistema</i>

2.4.1.4.7.8 Iniciar tarea del expediente

```
TrMensajeCondicionAccion[ ] iniciarTareaExpediente(
    TpoPK idExpediente,
    TpoPK idTareaFase,
    TpoPK idDefProc,
    String tipoTarea,
    TpoDate fecha,
    TpoDate fechaLimite,
    TpoPK expFase,
```

String observaciones,
TpoPK idTareaExp,
TrPlantillaTipoDocumento plantilla) throws TrException

Permite iniciar una nueva tarea para el expediente indicado en el parámetro *idExpediente*, dependiendo del valor del atributo *comprobarBloqueo* se comprobará si esta bloqueado o no. El tipo de la tarea puede ser: **"I"**: Incorporar un documento al expediente; **"G"**: Generar un documento para el expediente; **"M"**: Tarea del tipo manipular datos; **"O"**: Otros tipos de tarea.

Si la plantilla se pasa como parámetro anula la consulta de plantilla del tipo de documento generándose dicha plantilla, de lo contrario generará la implementada en la lógica de la función asociada a la consulta en caso que este asociada y válida para el tipo de documento.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idTareaFase</i>	<i>Identificador de la tarea en fase Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String tipoTarea</i>	<i>Tipo de tarea → "I" – Incorporar → "G" – Generar → "M" – Manipular datos → "O" – Otras</i>
<i>TpoDate fecha</i>	<i>Fecha de inicio de la tarea Si se pasa 'null' por defecto se coge la del sistema</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite para realizar la tarea</i>
<i>TpoPK expFase</i>	<i>Identificador de la instancia de la fase Si el identificador es 'null', el valor será la última instancia</i>
<i>String observaciones</i>	<i>Observaciones de la tarea</i>
<i>TpoPK idTareaExp</i>	<i>Identificador de la nueva tarea del expediente Entrada/Salida</i>
<i>TrPlantillaTipoDocumento plantilla</i>	<i>La plantilla del tipo de documento.</i>

Salida

TrMensajeCondicionAccion [] *Array de objetos TrMensajeCondicionAccion.*

TrMensajeCondicionAccion[] iniciarTareaExpediente(

*TpoPK idExpediente,
TpoPK idTareaFase,
TpoPK idDefProc,
String tipoTarea,
TpoDate fecha,
TpoDate fechaLimite,*

String observaciones,
TpoPK idTareaExp) throws TrException

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idTareaFase</i>	<i>Identificador de la tarea en fase Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String tipoTarea</i>	<i>Tipo de tarea → "I" – Incorporar → "G" – Generar → "M" – Manipular datos → "O" – Otras</i>
<i>TpoDate fecha</i>	<i>Fecha de inicio de la tarea Si se pasa 'null' por defecto se coge la del sistema</i>
<i>TpoDate fechaLimite</i>	<i>Fecha límite para realizar la tarea</i>
<i>String observaciones</i>	<i>Observaciones de la tarea</i>
<i>TpoPK idTareaExp</i>	<i>Identificador de la nueva tarea del expediente Entrada/Salida</i>

Salida

TrMensajeCondicionAccion [] Array de objetos *TrMensajeCondicionAccion*.

TrMensajeCondicionAccion[] iniciarTareaExpediente(
TpoPK idExpediente,
TpoPK idTareaFase,
TpoPK idDefProc,
String tipoTarea,
TpoDate fecha,
TpoDate fechaLimite,
TpoPK expFase,
String observaciones,
TpoPK idTareaExp) throws TrException

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idTareaFase</i>	<i>Identificador de la tarea en fase Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento Si el identificador es 'null', o no válido, se generará una excepción</i>

<i>String</i> tipoTarea	<i>Tipo de tarea</i> → “ I ” – Incorporar → “ G ” – Generar → “ M ” – Manipular datos → “ O ” – Otras
<i>TpoDate</i> fecha	<i>Fecha de inicio de la tarea</i> Si se pasa 'null' por defecto se coge la del sistema
<i>TpoDate</i> fechaLimite	<i>Fecha límite para realizar la tarea</i>
<i>TpoPK</i> expFase	<i>Identificador de la instancia de la fase</i>
<i>String</i> observaciones	<i>Observaciones de la tarea</i>
<i>TpoPK</i> idTareaExp	<i>Identificador de la nueva tarea del expediente</i> Entrada/Salida

Salida

TrMensajeCondicionAccion [] Array de objetos *TrMensajeCondicionAccion*.

2.4.1.4.7.9 Modificar tarea del expediente

```
void modificarTareaExpediente(
    TpoPK idTareaExp,
    String tipoTarea,
    TpoDate fechalnicio,
    TpoDate fechaLimite,
    String observaciones) throws TrException
```

Permite modificar la fecha de inicio, fecha limite y observaciones de una tarea de un expediente. El tipo de la tarea puede ser: “**I**”: Para un documento incorporado; “**G**”: Para un documento generado; “**M**”: Tarea del tipo manipular datos; “**O**”: Otros tipos de tarea.

Entradas

<i>TpoPK</i> idTareaExp	<i>Identificador de la tarea</i> Si el identificador es 'null', o no válido, se generará una excepción
<i>String</i> tipoTarea	<i>Tipo de tarea</i> → “ I ” – Incorporar → “ G ” – Generar → “ M ” – Manipular datos → “ O ” – Otras
<i>TpoDate</i> fechalnicio	<i>Fecha de inicio de la tarea</i> Si se pasa 'null' por defecto se coge la del sistema
<i>TpoDate</i> fechaLimite	<i>Fecha límite de realización de la tarea</i>
<i>String</i> observaciones	<i>Observaciones de la tarea</i>

2.4.1.4.7.10 Bloques definidos

```
TrBloque[ ] obtenerBloquesDefinidos(
    TpoPK idBloque,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException
```

Devuelve los datos sobre los bloques o módulos detalles definidos en el tramitador. Si se indica el parámetro *idBloque* se devolverán los datos del bloque correspondiente, si no (es *null*) se devuelven todos.

Entradas

<i>TpoPK idBloque</i>	<i>Identificador del bloque. Si es null, devuelve los datos correspondientes a todos los bloques.</i>
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrBloque.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrBloque.

Salida

TrBloque []	Array de objetos TrBloque con la información de bloques.
--------------	--

2.4.1.4.7.11 Bloques de datos permitidos en una fase

<p>TrBloquePermitido[] obtenerBloquesPermitidos(String bloqueLlamante, TpoPK idFase, TpoPK idDefProc, TpoPK idExpediente, TpoDate fecha, String condVisual, boolean soloUsuario, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Devuelve al usuario el conjunto de datos sobre los bloques o módulos detalles para los que se tiene acceso desde una fase dada y para un procedimiento también dado, desde un bloque que también pasamos como parámetro (el nombre) y que representa el bloque desde el cual se puede llamar.

Además recibe los parámetros *idExpediente* y *fecha* para evaluar las condiciones. El parámetros *condVisual* permite indicar si se obtienen los bloques que cumplan las condiciones de visualización ("S"), las que no ("N") o todas ("T") y el parámetro *soloUsuario* permite filtrar por el usuario establecido en el api y su perfil.

Entradas

<i>String bloqueLlamante</i>	<i>Nombre del bloque llamante.</i>
<i>TpoPK idFase</i>	<i>Identificador de la fase. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrBloquePermitido.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrBloquePermitido.

Salida

TrBloquePermitido [] Array de objetos TrBloquePermitido con la información de bloques permitidos.

TrBloquePermitido[] obtenerBloquesPermitidos(
String bloqueLlamante,
TpoPK idFase,
TpoPK idDefProc,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException *Deprecado*

Entradas

String bloqueLlamante Nombre del bloque llamante.
TpoPK idFase Identificador de la fase.
Si el identificador es 'null', o no válido, se generará una excepción
TpoPK idDefProc Identificador de la definición del procedimiento.
Si el identificador es 'null', o no válido, se generará una excepción
ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrBloquePermitido.
ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrBloquePermitido.

Salida

TrBloquePermitido [] Array de objetos TrBloquePermitido con la información de bloques permitidos.

TrBloquePermitido[] obtenerBloquesPermitidos(
String bloqueLlamante,
TpoPK idFase,
TpoPK idDefProc,
TpoPK idExpediente,
TpoPK idExpFase,
TpoDate fecha,
String condVisual,
boolean soloUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Entradas

<i>String</i> bloqueLlamante	Nombre del bloque llamante.
<i>TpoPK</i> idFase	Identificador de la fase. Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoPK</i> idDefProc	Identificador de la definición del procedimiento. Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoPK</i> idExpediente	Identificador del expediente Opcional
<i>TpoPK</i> idExpFase	Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.
<i>TpoDate</i> fecha	Fecha Si se pasa 'null', por defecto se coge la del sistema.
<i>String</i> condVisual	Indica qué bloques se obtienen → "S" – Las que cumplen las condiciones de visualización → "N" – Las que no cumplen las condiciones de visualización → "T" – Todos
<i>boolean</i> soloUsuario	Indica si se filtra por el usuario establecido en el api y su perfil
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrBloquePermitido.
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrBloquePermitido.

Salida

<i>TrBloquePermitido</i> []	Array de objetos TrBloquePermitido con la información de bloques permitidos.
------------------------------	--

2.4.1.4.7.12 Parámetros de un bloque de datos

<p>TrParametroBloque[] obtenerParametrosBloques(<i>TpoPK</i> idBloque, <i>ClausulaWhere</i> where, <i>ClausulaOrderBy</i> orderBy) throws TrException</p>
--

Devuelve al usuario el conjunto de parámetros definido para un bloque dado.

Entradas

<i>TpoPK</i> idBloque	Identificador del bloque.
-----------------------	---------------------------

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrParametroBloque.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrParametroBloque.

Salida

TrParametroBloque [] Array de objetos *TrParametroBloque*.

2.4.1.4.8 APIs sobre caducidades

2.4.1.4.8.1 Ampliar o reducir el plazo para una caducidad del expediente

void ampliarReducirCaducidadExpediente(

 TpoPK idCaducidadExp,
 String tipo,
 String unidad,
 int numUnidades,
 TpoDate fecha) throws TrException

Permite modificar una caducidad del expediente, ampliando o reduciendo el plazo.

Entradas

<i>TpoPK idCaducidadExp</i>	<i>Identificador de la caducidad del expediente. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String tipo</i>	<i>Tipo de la modificación. → "A" – Ampliación → "R" – Reducción</i>
<i>String unidad</i>	<i>Unidad de medida del tiempo. → "D" – Días → "M" – Meses → "A" – Años</i>
<i>Int numUnidades</i>	<i>Número de unidades que se cuentan.(nº de días, meses, años).</i>
<i>TpoDate fecha</i>	<i>Fecha en la que se produjo la modificación sobre la caducidad o el aviso. Si este parámetro no se indica se toma la fecha del sistema</i>

2.4.1.4.8.2 Establecer avisos sobre una caducidad del expediente

```

void establecerAvisoCaducidadExpediente(
    TpoPK idCaducidadExp,
    TpoDate fecha,
    TpoDate fechaAviso,
    TpoPK idTransicion,
    String usuario) throws TrException
  
```

Permite definir un aviso previo, en transición que se hace o en tiempo para la caducidad de un expediente.

Entradas

<i>TpoPK idCaducidadExp</i>	<i>Identificador de la caducidad del expediente. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoDate fecha</i>	<i>Fecha en la que se da de alta el aviso. Si no se indica, toma la fecha del sistema.</i>
<i>TpoDate fechaAviso</i>	<i>Fecha en la que se quiere que se avise.</i>
<i>TpoPK idTransicion</i>	<i>Identificador de la transición en la que hay que avisar al usuario. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String usuario</i>	<i>Usuario al que hay que avisar.</i>

Para establecer el aviso se debe indicar *fechaAviso* o *idTransicion* y obligatoriamente *usuario*.

Si se indican tanto *fechaAviso* como *idTransicion* el aviso se entenderá como lo que se produzca primero, es decir, o bien llegue la fecha de aviso o bien en el expediente se produzca la transición.

2.4.1.4.8.3 Avisos establecidos en la caducidad del expediente

```

TrAvisoCaducidad[] obtenerAvisosCaducidadExpediente(
    TpoPK idCaducidadExp,
  
```

ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve los avisos definidos sobre una caducidad de un expediente.

Entradas

<i>TpoPK</i> idCaducidadExp	<i>Identificador de la caducidad del expediente. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>ClausulaWhere</i> where	<i>Filtro a aplicar. Consultar campos posibles para TrAvisoCaducidad.</i>
<i>ClausulaOrderBy</i> orderBy	<i>Ordenación a aplicar. Consultar campos posibles para TrAvisoCaducidad.</i>

Salida

<i>TrAvisoCaducidad</i> []	<i>Array de objetos TrAvisoCaducidad.</i>
-----------------------------	---

2.4.1.4.8.4 Caducidades definidas

TrCaducidad[] obtenerCaducidadesDefProcedimiento (
TpoPK idDefProc,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve las caducidades definidas sobre un procedimiento concreto que se pasa como parámetro.

Entradas

<i>TpoPK</i> idStma	<i>Identificador de la definición del procedimiento. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>ClausulaWhere</i> where	<i>Filtro a aplicar. Consultar campos posibles para TrCaducidad.</i>
<i>ClausulaOrderBy</i> orderBy	<i>Ordenación a aplicar. Consultar campos posibles para TrCaducidad.</i>

Salida

<i>TrCaducidad</i> []	<i>Array de objetos TrCaducidad.</i>
------------------------	--------------------------------------

2.4.1.4.8.5 Caducidades activas del expediente

TrCaducidadExpediente[] obtenerCaducidadesExpediente (
TpoPK idExpediente,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve las caducidades activas para un expediente.

Entradas

<i>TpoPK</i> idExpediente	<i>Identificador del expediente. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>ClausulaWhere</i> where	<i>Filtro a aplicar. Consultar campos posibles para TrCaducidadExpediente.</i>

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrCaducidadExpediente.

Salida

TrCaducidadExpediente [] Array de objetos TrCaducidadExpediente.

2.4.1.4.8.6 Obtención de los expedientes caducados

TrExpedienteCaducado[] obtenerExpedientesCaducados(
TpoPK idStma,
TpoDate fechaReferencia,
String tipo,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve al usuario para un sistema dado (sino se indica sistema de devuelven todos), el conjunto de expedientes que estarían caducados a la fecha de referencia indicada (si no se indica se toma la del sistema).

El expediente puede haber caducado bien por una de las caducidades definidas en el flujo, o bien porque ha superado la fecha límite de estancia en una fase.

Entradas

<i>TpoPK</i> idStma	<i>Identificador del sistema.</i> <i>Opcional</i>
<i>TpoDate</i> fechaReferencia	<i>Fecha de referencia</i>
String tipo	<i>Tipo de caducidad que se quiere buscar.</i> → “ C ” – Caducidades definidas en el flujo → “ F ” – Caducidades de fase → “ A ” – Ambas
<i>ClausulaWhere</i> where	<i>Filtro a aplicar.</i> Consultar campos posibles para TrExpedienteCaducado. → Filtros sobre CAMPO_REFSTMA no tendrán efecto si el método recibe un identificador de sistema distinto de 'null'. → Filtros sobre CAMPO_REFCADEXP no tendrán efecto si el tipo de filtro es “F”. → Filtros sobre CAMPO_REFFASE no tendrán efecto si el tipo de filtro es “C”.
<i>ClausulaOrderBy</i> orderBy	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrExpedienteCaducado.

Salida

TrExpedienteCaducado [] Array de objetos TrExpedienteCaducado.

→ Cuando el expediente ha caducado debido a una de las caducidades definidas en el flujo, se devuelve CADUCIDADEXP.REFCADEXP pero si ha caducado por haber superado el tiempo límite en una fase se devolverá REFFASE.

TrExpedienteCaducado[] obtenerExpedientesCaducados(
TpoPK idStma,

<p>TpoDate fechaDesde, TpoDate fechaHasta, String tipo, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>

Devuelve al usuario para un sistema dado, el conjunto de expedientes que estarían caducados entre las fechas de referencia indicadas. Si no especifica el sistema, se devuelven todos.

Devolverá los expedientes caducados cuya fecha de caducidad sea mayor o igual al parámetro fechaDesde y menor que la indicada en el parámetro fechaHasta.

El expediente puede haber caducado bien por una de las caducidades definidas en el procedimiento, o bien porque ha superado la fecha límite de estancia en una fase.

Entradas

<i>TpoPK idStma</i>	<i>Identificador del sistema. Opcional</i>
<i>TpoDate fechaDesde</i>	<i>Fecha desde la que se quiere consultar los expedientes caducados</i>
<i>TpoDate fechaHasta</i>	<i>Fecha hasta la que se quiere consultar los expedientes caducados</i>
<i>String tipo</i>	<i>Tipo de caducidad que se quiere buscar. → "C" – Caducidades definidas en el flujo → "F" – Caducidades de fase → "A" – Ambas</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrExpedienteCaducado. → Filtros sobre CAMPO_REFSTMA no tendrán efecto si el método recibe un identificador de sistema distinto de 'null'. → Filtros sobre CAMPO_REFCADEXP no tendrán efecto si el tipo de filtro es "F". → Filtros sobre CAMPO_REFFASE no tendrán efecto si el tipo de filtro es "C".</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrExpedienteCaducado.</i>

Salida

<i>TrExpedienteCaducado []</i>	<i>Array de objetos TrExpedienteCaducado. → Cuando el expediente ha caducado debido a una de las caducidades definidas en el flujo, se devuelve CADUCIDADEXP.REFCADEXP pero si ha caducado por haber superado el tiempo límite en una fase se devolverá REFFASE. → La fecha de caducidad sea de una caducidad definida o por haber superado el tiempo límite en una fase se devuelve en CADUCIDADEXP.FECHALIMITE.</i>
---------------------------------	--

2.4.1.4.8.7 Obtención de días festivos

<p>TrDiaFestivo[] obtenerDiasFestivos(TpoPK idExpediente,</p>
--

Timestamp fechaInicial,
Timestamp fechaFinal) throws TrException

Devuelve una lista de días festivos comprendidos entre la fechaInicial la fechaFinal. Los días festivos se obtienen a partir de la provincia y municipio del organismo al que pertenece el expediente.

Entradas

<i>TpoPK</i> idExpediente	<i>Identificador del expediente</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>Timestamp</i> fechaInicial	<i>Fecha inicial</i>
<i>Timestamp</i> fechaFinal	<i>Fecha final</i>

Salida

TrDiaFestivo [] *Array de objetos TrDiaFestivo.*

TrDiaFestivo[] obtenerDiasFestivos(
String codProvincia,
String codMunicipio,
Timestamp fechaInicial,
Timestamp fechaFinal) throws TrException

Devuelve una lista de días festivos comprendidos entre la fechaInicial la fechaFinal. Los días festivos se obtienen a partir de la provincia y municipio indicados.

Entradas

<i>String</i> codProvincia	<i>Identificador de la provincia.</i> Si se indica null se obtendrán los festivos nacionales.
<i>String</i> codMunicipio	<i>Identificador del municipio.</i> Si se indica el municipio la provincia es obligatoria. Si no se indica se devolverán los festivos provinciales (si se ha indicado provincia) ó nacionales si no se ha indicado ninguno.
<i>Timestamp</i> fechaInicial	<i>Fecha inicial</i>
<i>Timestamp</i> fechaFinal	<i>Fecha final</i>

Salida

TrDiaFestivo [] *Array de objetos TrDiaFestivo.*

2.4.1.4.8 Modificaciones en la caducidad del expediente

TrModificacionCaducidadExpediente[] obtenerModificacionesCaducidadExpediente(
TpoPK idCaducidadExp,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve las modificaciones realizadas sobre una caducidad concreta de un expediente.

Entradas

<i>TpoPK idCaducidadExp</i>	<i>Identificador de la caducidad del expediente.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrModificacionCaducidadExpediente.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrModificacionCaducidadExpediente.

Salida

TrModificacionCaducidadExpediente [] *Array de objetos TrModificacionCaducidadExpediente.*

→ *Para el caso de "suspensiones" de tiempo, no tiene sentido devolver la información de los atributos UNIDAD y NUMUNIDADES. En este caso sólo se devolverá FECHA (en la que se produce la suspensión) y FECHAFINAL (en la que se reanuda la cuenta de tiempo).*

Para las modificaciones tipo "S", si esta fecha final es nula indicará que sobre la caducidad se ha suspendido el plazo y aún no se ha reanudado.

Sólo podrá existir una modificación "viva" del tipo "S", es decir, sólo existirá una con fecha final nula.

2.4.1.4.9 APIs sobre el sistema de intercambio de mensajes

2.4.1.4.9.1 Crear comunicación general

```
void crearComunicacionGeneral(
    TpoPK idDefProc, String titulo, String texto, String prioridad, TpoDate fechaInicio,
    TpoDate fechaFin) throws TrException
```

Permite a un usuario crear una comunicación a un interesado de un expediente indicado.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador de procedimiento al que se va enviar la comunicación.</i> Si el identificador de procedimiento es 'null', o no válido, se generará una excepción.
<i>String titulo</i>	<i>Título que se asociará a la comunicación personal.</i>
<i>String texto</i>	<i>Texto con el mensaje que contiene la comunicación.</i>

<i>String prioridad</i>	<i>Prioridad que tiene la comunicación personal.</i> → 'A' – Prioridad alta → 'M' – Prioridad media → 'B' – Prioridad baja
<i>TpoDate fechalnicio</i>	<i>Fecha de inicio de vigencia de la comunicación.</i>
<i>TpoDate fechaFin</i>	<i>Fecha de fin de vigencia de la comunicación.</i>

2.4.1.4.9.2 Crear comunicación personal

```
void crearComunicacionPersonal(
    TrInteresadoExpediente interesadoExp, String titulo, String texto, String prioridad,
    TpoDate fechalnicio, TpoDate fechaFin) throws TrException
```

Permite a un usuario crear una comunicación a un interesado de un expediente indicado.

Entradas

<i>TrInteresadoExpediente interesadoExp</i>	<i>Interesado expediente al que se va enviar la comunicación. Si el interesado expediente es 'null', o no válido, se generará una excepción.</i>
<i>String titulo</i>	<i>Título que se asociará a la comunicación personal.</i>
<i>String texto</i>	<i>Texto con el mensaje que contiene la comunicación.</i>
<i>String prioridad</i>	<i>Prioridad que tiene la comunicación personal.</i> → 'A' – Prioridad alta → 'M' – Prioridad media → 'B' – Prioridad baja
<i>TpoDate fechalnicio</i>	<i>Fecha de inicio de vigencia de la comunicación.</i>
<i>TpoDate fechaFin</i>	<i>Fecha de fin de vigencia de la comunicación.</i>

2.4.1.4.9.3 Enviar un mensaje a un usuario

```
void crearMensaje(
    String texto,
    String prioridad,
    String usuarioDestino,
    TpoPK idExpediente) throws TrException Deprecado
```

Permite al usuario crear una mensaje en el sistema de intercambio de mensaje. Permite indicar el texto, prioridad, el usuario destino y asociar el mensaje a un expediente.

Entradas

<i>String texto</i>	<i>Texto del mensaje</i>
<i>String prioridad</i>	<i>Indica la prioridad del mensaje</i> → 'A' – Prioridad alta → 'M' – Prioridad media → 'B' – Prioridad baja

String usuarioDestino

Usuario receptor del mensaje

TpoPK idExpediente

*Identificador del expediente asociado al mensaje.
Si el identificador es 'null', o no válido, se generará una excepción*

void crearMensaje(

*String texto,
String prioridad,
String usuarioDestino,
TpoPK idPerfilUsuario,
TpoDate fecha,
TpoPK idExpediente)* throws *TrException*

Entradas

String texto

Texto del mensaje

String prioridad

Indica la prioridad del mensaje

→ **'A'** – Prioridad alta

→ **'M'** – Prioridad media

→ **'B'** – Prioridad baja

String usuarioDestino

Usuario receptor del mensaje

TpoPK idPerfilUsuario

Identificador del perfil de usuario. El mensaje se enviará a los usuarios que tengan asociado este perfil de usuario, si uno de ellos es el usuarioDestino solo se enviará una vez. Opcional si se indica usuarioDestino.

TpoDate fecha

Fecha del mensaje

TpoPK idExpediente

*Identificador del expediente asociado al mensaje.
Si el identificador es 'null', o no válido, se generará una excepción*

2.4.1.4.9.4 Eliminar un mensaje

void eliminarMensaje(

TpoPK idMensaje) throws *TrException*

Permite a un usuario la eliminación de los mensajes recibidos en el sistema de intercambio de mensajes.

Si el usuario es "administrador" puede eliminar cualquier mensaje ya sea suyo o no.

Entradas

TpoPK idMensaje

Identificador del mensaje.

Si el identificador es 'null', o no válido, se generará una excepción.

2.4.1.4.9.5 Modificar el estado de un mensaje

void modificarEstadoMensaje(

TpoPK idMensaje,
String leído) throws TrException

Permite al usuario modificar el estado de un mensaje recibido, permite poner un mensaje no leído como leído. Un usuario "Administrador" puede modificar el estado de todos los mensajes, ya sean suyos o no.

Entradas

<i>TpoPK idMensaje</i>	<i>Identificador del mensaje.</i>
<i>String leído</i>	<i>Indica el estado del mensaje.</i> → 'S' – El mensaje ha sido leído → 'N' – El mensaje no ha sido leído

2.4.1.4.9.6 Modificar el estado de una comunicación

void modificarEstadoComunicacion(
TpoPK idComunicacion,
TpoDate fecha) throws TrException

Permite modificar el estado de una comunicación, poniéndola en estado "leída".

Entradas

<i>TpoPK idComunicacion</i>	<i>Identificador de la comunicación.</i>
<i>TpoDate fecha</i>	<i>Fecha de lectura de la comunicación, en la que se pone el estado a "leído".</i>

2.4.1.4.9.7 Obtener comunicaciones personales

TrComunicacion[] obtenerComunicacionesPersonales(
TrInteresadoExpediente interesadoExp, ClausulaWhere where, ClausulaOrderBy
orderBy) throws TrException

Devuelve todas las comunicaciones personales de un interesado expediente indicado.

Entradas

<i>TrInteresadoExpediente interesadoExp</i>	<i>Interesado expediente del que se obtienen las comunicaciones. Si el interesado expediente es 'null', o no válido, se generará una excepción.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrComunicacion.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrComunicacion.</i>

Salida

TrComunicacion [] *Array de objetos TrComunicacion.*

2.4.1.4.9.8 Obtener comunicaciones generales

TrComunicacion[] obtenerComunicacionesGenerales(
 TpoPK idDefProc, ClausulaWhere where, ClausulaOrderBy orderBy) throws
 TrException

Devuelve todas las comunicaciones personales de un interesado expediente indicado.

Entradas

<i>TrInteresadoExpediente</i> <i>interesadoExp</i>	<i>Interesado expediente del que se obtienen las comunicaciones.</i> <i>Si el interesado expediente es 'null', o no válido, se generará una excepción.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> <i>Consultar campos posibles para TrComunicacion.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> <i>Consultar campos posibles para TrComunicacion.</i>

Salida

TrComunicacion [] *Array de objetos TrComunicacion.*

2.4.1.4.9.9 Mensajes enviados

TrMensaje[] obtenerMensajesEnviados(
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener el conjunto de mensajes que un usuario ha enviado mediante el sistema de intercambio de mensajes.

Entradas

<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> <i>Consultar campos posibles para TrMensaje.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> <i>Consultar campos posibles para TrMensaje.</i>

Salida



TrMensaje []

Array de objetos TrMensaje.

2.4.1.4.9.10 Mensajes de un usuario

TrMensaje[] obtenerMensajesUsuario(

ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener el conjunto de mensajes que un usuario tiene en el sistema de intercambio de mensajes.

Entradas

ClausulaWhere where

Filtro a aplicar.

Consultar campos posibles para TrMensaje.

ClausulaOrderBy orderBy

Ordenación a aplicar.

Consultar campos posibles para TrMensaje.

Salida

TrMensaje []

Array de objetos TrMensaje.

2.4.1.4.10 APIs sobre interesados

2.4.1.4.10.1 Eliminar interesado en un expediente

void eliminarInteresadoExpediente(

TpoPK idExpediente,
TpoPK idInteresado,
TpoPK idRazonInt,
boolean elimInteDocu) throws TrException

Permite eliminar los interesados en un expediente. Además si se indica en el parámetro *elimInteDoc* se eliminarán los registros hijos de interesados en un documento y sus notificaciones.

Si el sistema establecido tiene conexión al bus y el interesado tiene ciw@ (Código identificador w@ndA) se informará del borrado del interesado.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idInteresado</i>	<i>Identificador del interesado Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idRazonInt</i>	<i>Identificador de la razón de interés Opcional</i>
<i>boolean elimInteDoc</i>	<i>Indica si se eliminan los interesados en el documento y sus notificaciones</i>

2.4.1.4.10.2 Eliminar límite de una relación entre interesados

void eliminarLimiteRelacion(TpoPK idLimiteRela) throws TrException

Permite eliminar un límite de la relación entre interesados identificada por idLimiteRela.

Entradas

<i>TpoPK idLimiteRela</i>	<i>Identificador del límite de la relación entre interesados Si el identificador es 'null', o no válido se generará una excepción</i>
---------------------------	---

2.4.1.4.10.3 Eliminar registro operación

void eliminarRegistroOperacion(TpoPK idRegistro) throws TrException

Permite eliminar un registro de una operación de una relación entre interesados y/o sus límites.

Entradas

<i>TpoPK idRegistro</i>	<i>Identificador registro de una operación de una relación entre interesados y/o sus límites Si el identificador es 'null', o no válido se generará una excepción</i>
-------------------------	---

2.4.1.4.10.4 Eliminar relación entre interesados

void eliminarRelacionInteresado(TpoPK idRelacionInte) throws TrException

Permite eliminar la relación entre interesados identificada por idRelacionInte.

Entradas

<i>TpoPK idRelacionInte</i>	<i>Identificador de la relación entre interesados Si el identificador es 'null', o no válido se generará una excepción</i>
-----------------------------	--

void eliminarRelacionInteresado(TpoPK idInteIni, TpoPK idInteFin, TpoPK idTipoCont) throws TrException

Permite eliminar las relaciones entre los interesados indicando los identificadores de los interesados inicio, fin y el tipo de contacto. Además permite eliminar todas las relaciones de un interesado si sólo se indica el parámetro *idIntelni*.

Entradas

<i>TpoPK idIntelni</i>	<i>Identificador de un interesado Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idInteFin</i>	<i>Identificador del otro interesado Opcional</i>
<i>TpoPK idTipoCont</i>	<i>Identificador del tipo de contacto entre los interesados Opcional</i>

2.4.1.4.10.5 Obtener interesados en el expediente

TrInteresadoExpediente [] obtenerInteresadosExpediente(
TpoPK idExpediente,
TpoPK idInteresado,
TpoPK idRazonInt,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener un conjunto de interesados en un expediente que se indica mediante el parámetro *idExpediente*. Si no se indica dicho parámetro se devolverán los de todos los expedientes.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente Opcional</i>
<i>TpoPK idInteresado</i>	<i>Identificador del interesado Opcional</i>
<i>TpoPK idRazonInt</i>	<i>Identificador de la razón de interés Opcional</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrInteresadoExpediente.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrInteresadoExpediente.</i>

Salida

<i>TrInteresadoExpediente []</i>	<i>Array de objetos TrInteresadoExpediente</i>
-----------------------------------	--

2.4.1.4.10.6 Obtener razones de interés documento

TrRazonInteresDocumento [] obtenerRazonInteresDocumento(
TpoPK idDocPer,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve un conjunto de razones de interés asociadas a un documento.

Entradas

<i>TpoPK idDocPer</i>	<i>Identificador del documento permitido.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrRazonInteresDocumento.</i>

ClausulaOrderBy orderBy

Ordenación a aplicar.

Consultar campos posibles para TrRazonInteresDocumento.

Salida

TrRazonInteresDocumento []

Array de objetos TrRazonInteresDocumento

2.4.1.4.10.7(b) Insertar interesado sin datos jurídicos de empresa

TpoPK insertarInteresado(
TrInteresadoSimple interesadoSimple,
TrDatosContacto datosContactoDefecto) throws *TrException*

Permite insertar un nuevo interesado con sus datos de contacto por defecto. Si no se envían datos de contacto no se le asignará ninguno. Se valida el Tipo Identificador y el Número Identificador con la función "validacionTipIdentificador". No es posible insertar un interesado con el mismo T_IDENTIFICADOR que otro interesado. Una vez insertado devuelve el identificador del nuevo interesado. No es posible indicar los datos de representación jurídica, para lo que habrá que insertarlos como un nuevo interesado y crear la relación de representación entre ambos interesados.

Los campos obligatorios para el interesado son TIPOIDENT, NUMIDENT, APELLIDO1, APELLIDO2, NOMBRE y SEXO.

Si el sistema establecido tiene conexión al bus se informará del nuevo interesado, quedando recogido en trew@ su ciw@ (Código identificador w@ndA). Para informar un interesado la fecha de nacimiento del interesado, el número y código postal de los datos de contacto son obligatorios. El interesado se recogerá en w@ndA como ciudadano si el tipo de identificador del ciudadano es "F" (física) y empresa si es "J" (jurídica).

Entradas

TrInteresadoSimple
interesadoSimple

Interesado a insertar

TrDatosContacto
datosContactoDefecto

Datos de contacto por defecto para el interesado

Salida

TpoPK

Identificador del nuevo interesado

TpoPK insertarInteresado(
TrInteresado interesado,
TrDatosContacto datosContactoDefecto) throws *TrException*
Deprecado

Entradas

TrInteresado interesado

Interesado a insertar

TrDatosContacto
datosContactoDefecto

Datos de contacto por defecto para el interesado

Salida

TpoPK

Identificador del nuevo interesado

2.4.1.4.10.8 Insertar interesado en un expediente

```

void insertarInteresadoExpediente(
    TpoPK idExpediente,
    TpoPK idRazonInt,
    TrInteresado interesado,
    TrDatosContacto datosContacto,
    String observaciones) throws TrException Deprecado

```

Permite asignar un interesado a un expediente concreto. Si no existe el interesado se insertará un nuevo, en este caso los datos de contacto serán obligatorios. Se valida el Tipo Identificador y el Número Identificador con la función "validacionTipoIdentificador". En caso de que el interesado ya existiese, y el parámetro *datosContacto* no se envíe, se tomarán por defecto los que tenga dicho interesado por defecto.

Los campos obligatorios para el interesado son TIPOIDENT, NUMIDENT, APELLIDO1, APELLIDO2, NOMBRE y SEXO; y para el dato de contacto los campos obligatorios son CODMUNICIPIO, CODPROVINCIA, CODTIPOVIA y NOMBREVIA.

Si el sistema establecido tiene conexión al bus se informará del nuevo interesado en el expediente, además de los cambios de fases. Si el expediente no había sido informado previamente se informará con todos sus datos en este momento.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idRazonInt</i>	<i>Identificador de la razón de interés Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TrInteresado interesado</i>	<i>Interesado a insertar al expediente</i>
<i>TrDatosContacto datosContacto</i>	<i>Datos de contacto para el interesado en el expediente</i>
<i>String observaciones</i>	<i>Observaciones del interesado en el expediente</i>

```

void insertarInteresadoExpediente(
    TpoPK idExpediente,
    TpoPK idRazonInt,
    TrInteresado interesado,
    TrDatosContacto datosContacto,
    String observaciones,
    Boolean notifica) throws TrException

```

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador del expediente Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idRazonInt</i>	<i>Identificador de la razón de interés Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TrInteresado interesado</i>	<i>Interesado a insertar al expediente</i>
<i>TrDatosContacto datosContacto</i>	<i>Datos de contacto para el interesado en el expediente</i>



String observaciones

Observaciones del interesado en el expediente

Boolean notifica

Notificación telemática como medio preferente

- **true** - Si desea notificación telemática

- **false** - No desea notificación telemática

Si el identificador es 'null', o no válido se generará una excepción

void insertarInteresadoExpediente(

TpoPK idExpediente,
TpoPK idRazonInt,
TrInteresado interesado,
TrDatosContacto datosContacto,
String observaciones,
Boolean notifica,
Timestamp fechaBaja,
String motivoBaja) throws TrException

Entradas

TpoPK idExpediente

Identificador del expediente

Si el identificador es 'null', o no válido se generará una excepción

TpoPK idRazonInt

Identificador de la razón de interés

Si el identificador es 'null', o no válido se generará una excepción

TrInteresado interesado

Interesado a insertar al expediente

TrDatosContacto datosContacto

Datos de contacto para el interesado en el expediente

String observaciones

Observaciones del interesado en el expediente

Boolean notifica

Notificación telemática como medio preferente

- **true** - Si desea notificación telemática

- **false** - No desea notificación telemática

Si el identificador es 'null', o no válido se generará una excepción

Timestamp fechaBaja

Fecha de baja asociada al interesado de un expediente

String motivoBaja

Motivo de la baja del interesado en un expediente

void insertarInteresadoExpediente(

TpoPK idExpediente,
TpoPK idRazonInt,
TrInteresado interesado,
TrDatosContacto datosContacto,
String observaciones,
Boolean notifica,
Timestamp fechaBaja,
String motivoBaja,
TrDatosContacto datosContactoNotificacion) throws TrException

Entradas

<i>TpoPK</i> idExpediente	Identificador del expediente Si el identificador es 'null', o no válido se generará una excepción
<i>TpoPK</i> idRazonInt	Identificador de la razón de interés Si el identificador es 'null', o no válido se generará una excepción
<i>TrInteresado</i> interesado	Interesado a insertar al expediente
<i>TrDatosContacto</i> datosContacto	Datos de contacto para el interesado en el expediente
<i>String</i> observaciones	Observaciones del interesado en el expediente
<i>Boolean</i> notifica	Notificación telemática como medio preferente - true - Si desea notificación telemática - false - No desea notificación telemática Si el identificador es 'null', o no válido se generará una excepción
<i>Timestamp</i> fechaBaja	Fecha de baja asociada al interesado de un expediente
<i>String</i> motivoBaja	Motivo de la baja del interesado en un expediente
<i>TrDatosContacto</i> datosContactoNotificacion	Datos de contacto de notificación del interesado

2.4.1.4.10.9 Insertar límite de una relación entre interesados

void insertarLimiteRelacion(TrLimiteRelacion limiteRelacion) throws TrException

Permite insertar un nuevo límite para una relación entre interesados, de esta forma se puede particularizar en una metafase, fase o expediente concreto.

Entradas

TrLimiteRelacion limiteRelacion *Límite de una relación entre interesados*

2.4.1.4.10.10 Insertar registro operación

void insertarRegistroOperacion(TrRegistroOperacion registro) throws TrException

Permite insertar un nuevo registro de una operación de una relación entre interesados y/o sus límites.

Entradas

<i>TrRegistroOperacion</i> registro	<i>Registro de una operación de una relación entre interesados y/o sus límites</i>
-------------------------------------	--

2.4.1.4.10.11 Insertar relación entre interesados

void insertarRelacionInteresado(TrRelacionInteresado relacionInteresado) throws TrException

Permite insertar una nueva relación entre interesados. El identificador de la nueva relación se podrá obtener mediante el método gerREFRELACIONINTE().

Entradas

<i>TrRelacionInteresado</i> relacionInteresado	<i>Relación entre interesados. Si el identificador es 'null', o no válido se generará una excepción</i>
---	---

void insertarRelacionInteresado(
TpoPK idInteIni,
TpoPK idInteFin,
TpoPK idTipoCont) throws TrException

Permite insertar una nueva relación entre interesados. Además de los identificadores de los interesados se deberá indicar el identificador del tipo de contacto existente entre ellos.

Entradas

<i>TpoPK idInteIni</i>	<i>Identificador de un interesado Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idInteFin</i>	<i>Identificador del otro interesado Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idTipoCont</i>	<i>Identificador del tipo de contacto entre los interesados Si el identificador es 'null', o no válido se generará una excepción</i>

2.4.1.4.10.12(b) Modificar interesado sin datos jurídicos de empresa

void modificarInteresado(
TrInteresadoSimple interesadoSimple,
TrDatosContacto datosContactoDefecto) throws TrException
Deprecado

Permite modificar los datos del interesado y sus datos de contacto por defecto. Para ello se comprueba que exista el interesado (interesado.REFINTERESADO). Se valida el Tipo Identificador y el Número Identificador con la función "validacionTipoIdentificador". Posteriormente se comprueba si se recibe el parámetro *datosContactoDefecto*. Si no se reciben datos de contacto se inserta sólo el interesado sin tener en cuenta el atributo interesado.REFDATOCONT, y si se recibe datos de contacto, se asignan por defecto al interesado.

Si el sistema establecido tiene conexión al bus se informará de las modificaciones del interesado. Si el interesado no tiene ciw@ (Código de identificador w@ndA) se informará el interesado como nuevo.

Entradas

<i>TrInteresadoSimple</i> interesadoSimple	<i>Interesado a modificar</i>
<i>TrDatosContacto</i> datosContactoDefecto	<i>Datos de contacto por defecto para el interesado</i>

```

void modificarInteresado(
    TrInteresado interesado,
    TrDatosContacto datosContactoDefecto) throws TrException
Deprecado

```

Entradas

<i>TrInteresado</i> interesado	<i>Interesado a modificar</i>
<i>TrDatosContacto</i> datosContactoDefecto	<i>Datos de contacto por defecto para el interesado</i>

2.4.1.4.10.13 Modificar interesado en un expediente

```

void modificarInteresadoExpediente(
    TpoPK idExpediente,
    TpoPK idInteresado,
    TpoPK idRazonInt,
    TrDatosContacto datosContacto,
    TpoPK idRazonIntNueva,
    String observaciones) throws TrException
Deprecado

```

Permite modificar un interesado en un expediente, así como sus datos de contacto. Además permite modificar la razón de interés del interesado en el expediente, siempre que no exista como interesado en un documento.

Si el sistema establecido tiene conexión al bus se informará de los cambios en el interesado expediente.

Entradas

<i>TpoPK</i> idExpediente	<i>Identificador del expediente</i> <i>Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK</i> idInteresado	<i>Identificador del interesado</i> <i>Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK</i> idRazonInt	<i>Identificador de la razón de interés</i> <i>Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TrDatosContacto</i> datosContacto	<i>Datos de contacto para el interesado en el expediente</i>

TpoPK idRazonIntNueva *Identificador de la nueva razón de interés*
String observaciones *Observaciones del interesado en el expediente*

```

void modificarInteresadoExpediente(
    TpoPK idExpediente,
    TpoPK idInteresado,
    TpoPK idRazonInt,
    TrDatosContacto datosContacto,
    TpoPK idRazonIntNueva,
    String observaciones,
    Boolean notifica) throws TrException

```

Entradas

TpoPK idExpediente *Identificador del expediente*
Si el identificador es 'null', o no válido se generará una excepción

TpoPK idInteresado *Identificador del interesado*
Si el identificador es 'null', o no válido se generará una excepción

TpoPK idRazonInt *Identificador de la razón de interés*
Si el identificador es 'null', o no válido se generará una excepción

TrDatosContacto datosContacto *Datos de contacto para el interesado en el expediente*

TpoPK idRazonIntNueva *Identificador de la nueva razón de interés*
String observaciones *Observaciones del interesado en el expediente*

Boolean notifica *Notificación telemática como medio preferente*
 - **true** - *Si desea notificación telemática*
 - **false** - *No desea notificación telemática*
Si el identificador es 'null', o no válido se generará una excepción

```

void modificarInteresadoExpediente(
    TpoPK idExpediente,
    TpoPK idInteresado,
    TpoPK idRazonInt,
    TrDatosContacto datosContacto,
    TpoPK idRazonIntNueva,
    String observaciones,
    Boolean notifica,
    Timestamp fechaBaja,
    String motivoBaja) throws TrException

```

Entradas

TpoPK idExpediente *Identificador del expediente*
Si el identificador es 'null', o no válido se generará una excepción

TpoPK idInteresado *Identificador del interesado*
Si el identificador es 'null', o no válido se generará una excepción

TpoPK idRazonInt *Identificador de la razón de interés*
Si el identificador es 'null', o no válido se generará una excepción

TrDatosContacto datosContacto *Datos de contacto para el interesado en el expediente*

TpoPK idRazonIntNueva *Identificador de la nueva razón de interés*

String observaciones

Observaciones del interesado en el expediente

Boolean notifica

Notificación telemática como medio preferente

- **true** - Si desea notificación telemática

- **false** - No desea notificación telemática

Si el identificador es 'null', o no válido se generará una excepción

Timestamp fechaBaja

Fecha de baja del interesado en un expediente

String motivoBaja

Motivo de baja del interesado en un expediente

2.4.1.4.10.14 Modificar relación entre interesados

<p>void modificarRelacionInteresado(TrRelacionInteresado relacionInteresado) throws TrException</p>
--

Permite modificar una relación entre interesados ya existentes. Para modificar la relación es imprescindible que se indique el identificador REFRELACIONINTE.

Entradas

TrRelacionInteresado
relacionInteresado

Relación entre interesados a modificar

Si el identificador es 'null', o no válido se generará una excepción



void modificarRelacionInteresado(

TpoPK idInteIni,
TpoPK idInteFin,
TpoPK idTipoCont,
TpoPK idInteFinNuevo,
TpoPK idTipoContNuevo) throws TrException

Modifica una relación entre interesados. Permite modificar el interesado con el que se relaciona mediante el parámetro *idInteFinNuevo* así como el tipo de contacto entre ellos mediante el parámetro *idTipoContNuevo*.

Entradas

<i>TpoPK idInteIni</i>	<i>Identificador de un interesado Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idInteFin</i>	<i>Identificador del otro interesado Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idTipoCont</i>	<i>Identificador del tipo de contacto entre los interesados Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idInteFinNuevo</i>	<i>Identificador del otro interesado a relacionar Si el identificador es 'null', o no válido se generará una excepción</i>
<i>TpoPK idTipoContNuevo</i>	<i>Identificador del nuevo tipo de contacto entre los interesados Si el identificador es 'null', o no válido se generará una excepción</i>

2.4.1.4.10.15 Modificar límite de una relación entre interesados

void modificarLimiteRelacion(TrLimiteRelacion limiteRelacion) throws TrException

Permite modificar un límite ya existente para una relación entre interesados, de esta forma se puede particularizar en una metafase, fase o expediente concreto.

Entradas

<i>TrLimiteRelacion</i>	<i>limiteRelacion</i>	<i>Límite de una relación entre interesados</i>
-------------------------	-----------------------	---

2.4.1.4.10.16 Modificar registro operación

void modificarRegistroOperacion(TrRegistroOperacion registro) throws TrException

Permite modificar registro de una operación de una relación entre interesados y/o sus límites.

Entradas

<i>TrRegistroOperacion</i>	<i>registro</i>	<i>Registro de una operación de una relación entre interesados y/o sus límites</i>
----------------------------	-----------------	--

2.4.1.4.10.17 Obtener datos de contacto del interesado

TrDatosContacto[] obtenerDatosContactoInteresado(
TpoPK idInteresado,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve los datos de contacto de un interesado.

Entradas

<i>TpoPK idInteresado</i>	<i>Identificador del interesado</i> <i>Si el identificador es 'null', o no válido se generará una excepción</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> <i>Consultar campos posibles para TrDatosContacto.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> <i>Consultar campos posibles para TrDatosContacto.</i>

Salida

TrDatosContacto [] *Array de objetos TrDatosContacto.*

2.4.1.4.10.18 Obtener interesados sin datos jurídicos de empresa

TrInteresadoSimple[] obtenerInteresados(
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException *Deprecado*

Permite obtener un conjunto de interesados simples, que no poseen los datos jurídicos, datos de empresa.

Entradas

<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> <i>Consultar campos posibles para TrInteresado.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> <i>Consultar campos posibles para TrInteresado.</i>

Salida

TrInteresadoSimple [] *Array de objetos TrInteresadoSimple.*
El atributo OTROSDATOS nos indica si el interesado tiene recogido otros datos – “S” – ó no los tiene - “N”.

TrInteresadoSimple[] obtenerInteresados(TpoPK idInte,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Entradas

<i>TpoPK idInte</i>	<i>Identificador del interesado</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> <i>Consultar campos posibles para TrInteresado.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> <i>Consultar campos posibles para TrInteresado.</i>

Salida

TrInteresadoSimple []

Array de objetos TrInteresadoSimple.

El atributo OTROSDATOS nos indica si el interesado tiene recogido otros datos – “S” – ó no los tiene - “N”.

2.4.1.4.10.19 Obtener interesados CC

TrInteresadoSimple[] obtenerInteresadosExpedienteCC(

String docIdentidad,
String apellido1,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Obtener para un interesado, obtener sus expedientes que pertenezcan a procedimientos integrados con CC, incluyendo su estado (E01: Abierto, E02: Finalizado).

Entradas

String docIdentidad

Documento de identidad del interesado

String apellido1

Primer apellido del interesado

ClausulaWhere where

Filtro a aplicar.
Consultar campos posibles para TrInteresado.

ClausulaOrderBy orderBy

Ordenación a aplicar.
Consultar campos posibles para TrInteresado.

Salida

TrInteresadoSimple []

Array de objetos TrInteresadoSimple.

El atributo OTROSDATOS nos indica si el interesado tiene recogido otros datos – “S” – ó no los tiene - “N”.

2.4.1.4.10.20 Obtener razones de interés

TrRazonInteres[] obtenerRazonesInteres(

ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve un conjunto de razones de interés.

Entradas

<i>ClausulaWhere</i> where	<i>Filtro a aplicar.</i> Consultar campos posibles para TrRazonInteres.
<i>ClausulaOrderBy</i> orderBy	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrRazonInteres.

Salida

TrRazonInteres [] *Array de objetos TrRazonInteres*

2.4.1.4.10.21 Obtener razones interés procedimiento

<p>TrRazonInteresProcedimiento [] obtenerRazonInteresProcedimiento(TpoPK idDefProc, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Permite obtener un conjunto de razones de interés del procedimiento.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador del procedimiento</i>
<i>ClausulaWhere</i> where	<i>Filtro a aplicar.</i> Consultar campos posibles para TrRazonInteres.
<i>ClausulaOrderBy</i> orderBy	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrRazonInteres.

Salida

TrRazonInteresProcedimiento [] *Array de objetos TrRazonInteresProcedimiento*

2.4.1.4.10.22 Obtener registros operación

<p>TrRegistroOperacion[] obtenerRegistroOperacion(TpoPK idRegistro, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Permite obtener un conjunto de registros de operaciones de relaciones entre interesados y sus límites.

Entradas

<i>TpoPK idRegistro</i>	<i>Identificador del límite</i> <i>Si el identificador es 'null', o no válido se devolverán todos</i>
<i>ClausulaWhere</i> where	<i>Filtro a aplicar.</i> Consultar campos posibles para TrLimiteRelacion.
<i>ClausulaOrderBy</i> orderBy	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrLimiteRelacion.

Salida

TrRegistroOperacion [] *Array de objetos TrRegistroOperacion*

2.4.1.4.10.23 Obtener relaciones entre interesados

<p>TrRelacionInteresado[] obtenerRelacionesInteresado(TpoPK idInteresado, ClausulaWhere where,</p>

ClausulaOrderBy orderBy) throws TrException

Permite obtener un conjunto de objetos TrRelacionInteresado que recoge todas las relaciones del interesado especificado en el parámetro *idInteresado* con otros interesados (sentido de la relación AB), así como las relaciones de otros interesados con el *idInteresado* (sentido de la relación BA).

Entradas

<i>TpoPK idInteresado</i>	<i>Identificador del interesado</i> <i>Si el identificador es 'null', o no válido se generará una excepción</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrRelacionInteresado.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrRelacionInteresado.

Salida

<i>TrRelacionInteresado []</i>	<i>Array de objetos TrRelacionInteresado</i>
---------------------------------	--

2.4.1.4.11Otras APIs

2.4.1.4.11.1Establece parámetros

void establecerConfiguracionSistema(
String report,
String conexion,
TpoPK idStma) throws TrException

Este procedimiento permite parametrizar las URL devueltas por el tramitador para abrir los documentos almacenados. Los valores indicados en los parámetros deben estar definidos antes como constantes en el tramitador.

Entradas

<i>String report</i>	<i>Cadena de llamada para el report server.</i>
<i>String conexion</i>	<i>Cadena de conexión que se suministra al report server.</i>
<i>TpoPK idStma</i>	<i>Identificador del sistema</i>

2.4.1.4.11.2 Obtención de datos de sistema

String getStrPerfil ()

Obtiene el nombre del sistema.

2.4.1.4.11.3 Activar cache para constantes

boolean isActivaCacheConstantes ()

Devuelve el estado actual de la caché para constantes.

2.4.1.4.11.4 Comprobación del formato de fecha

String obtenerFormatoFecha()

Devuelve el formato de fecha establecido en el TrAPIUI.

Salida

<i>String</i>	<i>Formato de fecha establecido.</i>
---------------	--------------------------------------

2.4.1.4.11.5 Obtener mensaje de error

String obtenerMensajeError(long numError) throws TrException;

Permite obtener el mensaje de error asociado al número de error que se indica en el parámetro numError.

Entradas

<i>long numError</i>	<i>Número del error</i>
----------------------	-------------------------

Salidas

<i>String</i>	<i>Cadena con el mensaje de error</i>
---------------	---------------------------------------

2.4.1.4.12 APIs exclusivas para Port@firmas

2.4.1.4.12.1 Actualiza entregas

void actualizaEntregas(TpoPK idDocExp) throws TrException

Método que comprueba las entregas en Port@firmas y actualiza la información en Trew@.

Desde Trew@ 1.5.0 se permite indicar mediante la propiedad “DESCARGA_PKCS7” para la configuración en Trew@ de Port@firmas (como dato del componente que se esté usando), para permitir indicar si así se desea la descarga o no del PKCS7 del documento desde @firma en el método “actualizaEntregas”. Sus valores pueden ser “TRUE” para que obligatoriamente realice la descarga del PKCS7, si se produce algún error se eleva la excepción, si el valor es “OPTIONAL” es realiza la llamada al servicio web pero si se produce un error no se eleva y se sigue la actualización y si el valor no es ninguno de los anteriores, por ejemplo es “FALSE” o “N”, no se realiza la llamada al servicio web. En determinados entornos el coste de esta operación suele ser elevado, esto permite mediante configuración obviar en dicho método del api el proceso de consulta de este atributo a @firma a través de port@firmas, si la aplicación no necesita del mismo almacenado en Trew@.

Si el estado es devuelto lanza una exception con el código -20420 en cuyo mensaje se indica el usuario que la la ha devuelto. Otras excepciones que puede lanzar este método son si la petición no existe (-20419) y si el documento no ha sido enviado a port@firmas (-20418).

Entrada

TpoPK idDocExp Identificador del documento del expediente a comprobar

void actualizaEntregas(
 TpoPK idDocExp,
 TpoPK idOrganizacion) **throws TrException**

Entrada

TpoPK idDocExp Identificador del documento del expediente a comprobar

TpoPK idOrganizacion Método que comprueba las entregas en Port@Firmas y actualiza la información en Trew@

2.4.1.4.12.2 Actualizar hash en petición de firma

void actualizarHashPeticionFirma(
 String hashPeticion,
 String idDocumento) **throws TrException**

Actualiza el hash de la petición de firma.

Entrada

<i>String</i> hashPetición	Hash de la petición
<i>String</i> idDocumento	Identificador del documento

2.4.1.4.12.3 Eliminar petición de Port@firmas

TrPeticiónEliminada[] eliminarPeticiónPortafirmas(TpoPK idDocExp, boolean multiple) throws TrException

Elimina la petición de firma del documento enviado a Port@firmas, siempre que el sistema establecido tenga asociado el componente de firma Port@firmas y que el documento haya sido enviado a Port@firmas.

Entrada

<i>TpoPK</i> idDocExp	Identificador del documento del expediente. Si es null se elevará una exception.
<i>boolean</i> multiple	Indica si el documento es múltiple y se quiere eliminar todas las peticiones de los documentos del mismo tipo realizados por el mismo usuario en una fecha determinada.

Salida

<i>TrPeticiónEliminada[]</i>	Devuelve un array de objetos TrPeticiónEliminada indicando el hash de la petición y un boolean que indica si se ha borrado (true) o no (false);
------------------------------	---

2.4.1.4.12.4 Existe usuario en Port@firmas

boolean existeUsuarioPortafirmas(String nif) throws TrException

Permite conocer si el usuario indicado mediante su nif/dni existe en el Port@firmas al que el sistema establecido envía los documentos para firmarlos digitalmente.

Entrada

<i>String</i>	Número de identificación del usuario, nif o dni.
---------------	--

Salida

<i>boolean</i>	Devuelve true si el usuario existe en Port@firmas y false en caso contrario.
----------------	--

2.4.1.4.12.5 Estados de la petición

String[] obtenerEstadosPortafirmas() throws TrException

Método que devuelve todos los estados posibles de una petición a Port@firmas

Salida

String[]

Array de objetos String's con los estados posibles de una transacción

2.4.1.4.13 APIs exclusivas para @visorador

El conjunto de apis exclusivas de @visorador sólo funcionarán si se ha definido el componente de @visorador en el sistema establecido en el TrAPIUI o en el sistema por defecto.

2.4.1.4.13.1 Crear aviso

```
void crearAviso(
    String asunto,
    String texto,
    String prioridad,
    int periodicidad,
    int repeticiones,
    String[] usuariosDestino,
    String[] gruposDestino,
    String[] emailsDestino) throws TrException;
```

Método que crea un aviso con los datos proporcionados

Entrada

<i>String</i> asunto	Título del aviso
<i>String</i> texto	Texto del aviso
<i>String</i> prioridad	Prioridad, (1 a 5), 0 o nulo valor por defecto
<i>int</i> periodicidad	Periodicidad, > 0 repeticiones cada X segundos
<i>int</i> repeticiones	Repeticiones del aviso
<i>String[]</i> usuariosDestino	Lista de usuarios de aplicación destinatarios
<i>String[]</i> gruposDestino	Lista de grupos a los que están suscritos los usuarios destinatarios
<i>String[]</i> emailsDestino	Direcciones de correo electrónico

2.4.1.4.13.2 Crear mensaje aviso

```
void crearMensajeAviso(
    String asunto,
    String texto,
    String prioridad,
    int periodicidad,
    int repeticiones,
    String usuarioDestino,
    TpoPK idExpediente) throws TrException;
```

Método que genera un aviso mediante @visorador y vincula este a un expediente

Entrada

<i>String</i> asunto	Título del aviso (solo aviso)
<i>String</i> texto	Contenido del mensaje/aviso
<i>String</i> prioridad	Prioridad del mensaje/aviso, (1 a 5), 0 o nulo valor por defecto
<i>int</i> periodicidad	Periodicidad del aviso (solo aviso)
<i>int</i> repeticiones	Veces que se remite el aviso (solo aviso)
<i>String</i>] usuarioDestino	Usuario destinatario del mensaje/aviso
<i>TpoPK</i>] idExpediente	Identificador del expediente al que se añade el mensaje

2.4.1.4.14 APIs Exclusivas para Notific@

El conjunto de apis exclusivas de notific@dor sólo funcionarán si se ha definido el componente de notific@dor en el sistema establecido en el TrAPIUI o en el sistema por defecto.

Otros datos de interesados, expedientes, documentos y procedimientos

String actualizarOtrosDatos (
TpoPK id,
String tipo, String contenido) throws TrException

Permite obtener otros datos de los interesados, expedientes, documentos o procedimientos según se indique en el parámetro *tipo*.

Entradas

<i>TpoPK id</i>	Identificador del interesado, expediente o procedimiento. Si el identificador es 'null' o no válido se generará una excepción.
<i>String tipo</i>	Indica el tipo del que se quieren obtener otros datos. → “ P ” – Procedimiento → “ I ” – Interesado → “ E ” – Expediente → “ D ” – Documento Si el tipo es 'null' o no válido se generará una excepción.
<i>String contenido</i>	Contenido para otros datos.

Salida

<i>String</i>	Otros datos del interesado, expediente o procedimiento.
---------------	---

String obtenerOtrosDatos(
TpoPK id,
String tipo) throws TrException

Permite obtener otros datos de los interesados, expedientes, documentos o procedimientos según se indique en el parámetro *tipo*.

Entradas

<i>TpoPK id</i>	Identificador del interesado, expediente o procedimiento. Si el identificador es 'null' o no válido se generará una excepción.
<i>String tipo</i>	Indica el tipo del que se quieren obtener otros datos. → “ P ” – Procedimiento → “ I ” – Interesado → “ E ” – Expediente → “ D ” – Documento Si el tipo es 'null' o no válido se generará una excepción.

Salida

<i>String</i>	Otros datos del interesado, expediente o procedimiento.
---------------	---

2.4.1.4.14.1 Notificar documentos interesados

void notificaDocumentosInteresados(

```
TrInteresadoDocumento[ ] documentosInteresados,
String asunto,
String texto) throws TrException; Deprecado
```

Método que para la lista de documentos por interesados, envía notificaciones a dichos usuarios con los documentos como adjuntos y con el asunto y texto pasado por parámetro.

Entrada

<i>TrInteresadoDocumento[]</i> documentosInteresados	Documentos y destinatarios
<i>String</i> asunto	Título de la notificación
<i>String</i> texto	Texto de la notificación

```
void notificaDocumentosInteresados(
    TrInteresadoDocumento[ ] documentosInteresados,
    String asunto,
    String texto,
    TrEmpleado empleado) throws TrException;
```

Entrada

<i>TrInteresadoDocumento[]</i> documentosInteresados	Documentos y destinatarios
<i>String</i> asunto	Título de la notificación
<i>String</i> texto	Texto de la notificación
<i>TrEmpleado</i> empleado	Empleado que realiza la notificación

2.4.1.4.14.2 Obtener estados abonados

```
Int[] obtenerEstadosAbonados (TpoPK[] idInteresados) throws TrException;
```

Método para obtener el estado de los interesados a un servicio de notificaciones.

Entrada

TpoPK[] idInteresados Identificadores de los interesados a consultar.
Si los identificadores no son válidos se generará una excepción

Int[] obtenerEstadosAbonados (*TpoPK[]* idInteresados, *TpoPK* idExpediente) throws
TrException;

Entrada

TpoPK[] idInteresados Identificadores de los interesados a consultar.
Si los identificadores no son válidos se generará una excepción

TpoPK idExpediente Identificador del expediente al que pertenecen los interesados a consultar.

2.4.1.4.14.3 Insertar notificación en un interesado

void insertarNotificacionInteresado (*TrNotificacionInteresado* notificacionInte) throws
TrException;

Permite insertar una notificación para un interesado en un documento.

Entrada

TrNotificacionInteresado *Notificación a insertar*
notificacionInte

2.4.1.4.15 Otras APIs de obtención de datos definidas

2.4.1.4.15.1 Activar cache para constantes

void setActivaCacheConstantes (boolean activaCacheConstantes)

Establece una caché para el almacenamiento de valores de constantes reduciéndose los accesos a las tablas de constantes de BBDD

Entrada

<i>boolean</i>	→ "true" – Se activa la caché una vez que se crea un api.
<i>activaCacheConstantes</i>	→ "false" – No se activa la caché cuando se ejecute un api.

void limpiarCacheConstantes ()

Método que limpia la caché de constantes. La caché se va recargando a medida que se vayan consultando los valores de las constantes, así sólo se produce un acceso a BBDD por constante.

2.4.1.4.15.2 Componentes

TrComponente[] obtenerComponentes(TpoPK idComponente, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Devuelve el componente que se indique en el parámetro *idComponente*. Si se envía 'null' se devuelven todos.

Entradas

<i>TpoPK idComponente</i>	Identificador del componente Si el identificador es 'null' devolverá todos los componentes
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrComponente.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrComponente.

Salida

<i>TrComponente []</i>	Array de objetos <i>TrComponente</i>
-------------------------	--------------------------------------

2.4.1.4.15.3 Constantes

TrConstante[] obtenerConstantes(TpoPK idSistema, String idConstante, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException;

Permite obtener un conjunto de constantes de los sistemas. Si se pasan alguno de los parámetros idConstante o idSistema se filtrará por ellos, si no se pasan se obtienen todos los registros.

Entradas

<i>TpoPK</i> idSistema	Identificador del sistema Si el identificador es 'null' se devolverán de todos los sistemas.
<i>String</i> idConstante	Identificador de la constante Si se indica y no existe elevará una excepción.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrConstante
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrConstante

Salida

<i>TrConstante</i> []	Array de objetos <i>TrConstante</i>
------------------------	-------------------------------------

2.4.1.4.15.4 Constantes generales

TrConstanteGn[] obtenerConstantesGenerales(String idConstanteGn, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException;

Permite obtener un conjunto de constantes generales en Trew@. Si no se indica el Identificador de la Constante se devolverán todas.

Entradas

<i>String</i> idConstanteGn	Identificador de la constante general Si el identificador es 'null' se devolverán todas las constantes
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrConstante
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrConstante

Salida

<i>TrConstanteGn</i> []	Array de objetos <i>TrConstanteGn</i>
--------------------------	---------------------------------------

2.4.1.4.15.5 Datos componentes

```
TrDatoComponente[ ] obtenerDatosComponente(
    TpoPK idDatoComponente,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener los datos de los componentes que se indique en el parámetro *idDatoComponente*. Si se envía 'null' se devuelven todos.

Entradas

<i>TpoPK idDatoComponente</i>	Identificador del dato del componente Si el identificador es 'null' devolverá todos
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrDatoComponente.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrDatoComponente.

Salida

<i>TrDatoComponente []</i>	Array de objetos <i>TrDatoComponente</i>
-----------------------------	--

2.4.1.4.15.6 Datos de una fase

```
TrFase[ ] obtenerDatosFase(
    TpoPK idFase,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException
```

Devuelve datos relativos a la fase cuya id es pasado como parámetro.

Entradas

<i>TpoPK idFase</i>	Identificador de la fase. Si el identificador es 'null' el método devolverá todas las fases.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrFase.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrFase.

Salida

<i>TrFase []</i>	Array de objetos TrFase con la información de fases.
-------------------	--

2.4.1.4.15.7 Empleados

TrEmpleado[] obtenerEmpleados(
String usuario,
TpoPK idOrganismo,
String idPtoTrab,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos del empleado que se indique por parámetros. Si se pasan a 'null' se devolverán todos.

Entradas

<i>String</i> usuario	Identificador del usuario Si el identificador es 'null' devolverá todos
<i>TpoPK</i> idOrganismo	Identificador del organismo Si el identificador es 'null' devolverá todos
<i>String</i> idPtoTrab	Identificador del puesto de trabajo Si el identificador es 'null' devolverá todos
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrEmpleado.
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrEmpleado.

Salida

<i>TrEmpleado</i> []	Array de objetos <i>TrEmpleado</i>
-----------------------	------------------------------------

2.4.1.4.15.8 Número de empleados

int obtenerEmpleadosCount(
String usuario,
TpoPK idOrganismo,
String idPtoTrab,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve el número de empleados.

Entradas

<i>String</i> usuario	Identificador del usuario Si el identificador es 'null' devolverá todos
<i>TpoPK</i> idOrganismo	Identificador del organismo Si el identificador es 'null' devolverá todos
<i>String</i> idPtoTrab	Identificador del puesto de trabajo Si el identificador es 'null' devolverá todos
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrEmpleado.
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrEmpleado.

Salida

TrEmpleado []

Array de objetos *TrEmpleado*

2.4.1.4.15.9 Fases

TrFase[] obtenerFases(TpoPK idFase
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener un conjunto de fases definidas en Trew@.

Entradas

<i>TpoPK</i> IdFase	Identificador de la fase. Si no se indica se devuelven todas. Si se indica se comprueba que sea una fase existente.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrFase.
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrFase.

Salida

TrFase [] Array de objetos TrFase

2.4.1.4.15.10 Número de fases

TrFase[] obtenerFasesCount(TpoPK idFase
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve el número de fases.

Entradas

<i>TpoPK</i> IdFase	Identificador de la fase. Si no se indica se devuelven todas. Si se indica se comprueba que sea una fase existente.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrFase.
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrFase.

Salida

TrFase [] Array de objetos TrFase

2.4.1.4.15.11 Obtener metadato



TrMetadato[] obtenerMetadato(
TpoPK idMetadato,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Registra un nuevo metadato para un sistema. Si todo es correcto devuelve el identificador del metadato, si algo falla lanza una excepción.

Entradas

TpoPK idMetadato Identificador del metadato a obtener

Entrada / Salida

TrMetadato[] Metadatos Array de metadatos obtenidos

2.4.1.4.15.12 Metafases

TrMetafase[] obtenerMetafases(TpoPK idMetafase
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener un conjunto de metafases definidas en Trew@.

Entradas

TpoPK idMetafase Identificador de la metafase. Si no se indica se devuelven todas.
Si se indica se comprueba que sea una metafase existente.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrMetafase.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrMetafase.

Salida

TrMetafase [] Array de objetos TrMetafase

2.4.1.4.15.13 Número de metafases

int obtenerMetafasesCount(TpoPK idMetafase
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve el número de metafases

Entradas

<i>TpoPK idMetafase</i>	Identificador de la metafase. Si no se indica se devuelven todas. Si se indica se comprueba que sea una metafase existente.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrMetafase.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrMetafase.

Salida

<i>int</i>	Número de registros
------------	---------------------

2.4.1.4.15.14Municipios

TrMunicipio[] obtenerMunicipios(
String idMunicipio,
String idProvincia,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos del municipio que se indique por parámetros. Si se pasan a 'null' se devolverán todos.

Entradas

<i>String idMunicipio</i>	Identificador del municipio Si el identificador es 'null' devolverá todos
<i>String idProvincia</i>	Identificador de la provincia Si el identificador es 'null' devolverá todos
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrMunicipio.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrMunicipio.

Salida

<i>TrMunicipio []</i>	Array de objetos <i>TrMunicipio</i>
------------------------	-------------------------------------

2.4.1.4.15.15 Obtención de los organismos

TrOrganismo[] obtenerOrganismos(
 TpoPK idOrganismo,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos del organismo que se indique en el parámetro *idOrganismo*. Si se envía 'null' se devuelven todos.

Entradas

<i>TpoPK idOrganismo</i>	Identificador del organismo Si el identificador es 'null' devolverá todos
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrOrganismo.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrOrganismo.

Salida

<i>TrOrganismo []</i>	Array de objetos <i>TrOrganismo</i>
------------------------	-------------------------------------

2.4.1.4.15.16 Número de organismos

TrOrganismo[] obtenerOrganismosCount(
 TpoPK idOrganismo,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Devuelve el número de organismos.

Entradas

<i>TpoPK idOrganismo</i>	Identificador del organismo Si el identificador es 'null' devolverá todos
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrOrganismo.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrOrganismo.

Salida

<i>TrOrganismo []</i>	Array de objetos <i>TrOrganismo</i>
------------------------	-------------------------------------

2.4.1.4.15.17 Organismos

TrOrganismo[] obtenerOrganismosHijos(
TpoPK idOrganismoPadre,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos de los organismos hijos del organismo que se indique en el parámetro *idOrganismo*.

Entradas

<i>TpoPK idOrganismoPadre</i>	Identificador del organismo padre.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrOrganismo.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrOrganismo.

Salida

<i>TrOrganismo []</i>	Array de objetos <i>TrOrganismo</i>
------------------------	-------------------------------------

2.4.1.4.15.18 Países

TrPais[] obtenerPaises(
String idPais,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos del país que se indique por parámetro. Si se pasa 'null' se devolverán todos.

Entradas

<i>String idPais</i>	Identificador del país Si el identificador es 'null' devolverá todos
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrPais.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrPais.

Salida

<i>TrPais []</i>	Array de objetos <i>TrPais</i>
-------------------	--------------------------------

2.4.1.4.15.19 Perfiles

TrPerfilUsuario[] obtenerPerfiles(TpoPK idSistema, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener los distintos perfiles definidos para los usuarios en un sistema indicado por parámetros. Si no se indica el parámetro idSistema se devolverán todos.

Entradas

<i>TpoPK idSistema</i>	Identificador del sistema. Si no se indica se devolverán todos los perfiles de todos los sistemas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrPerfilUsuario.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrPerfilUsuario.

Salida

<i>TrPerfilUsuario []</i>	Array de objetos <i>TrPerfilUsuario</i>
----------------------------	---

2.4.1.4.15.20 Perfiles del usuario

TrPerfilUsuario[] obtenerPerfilesUsuario(ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener los distintos perfiles de usuario que tiene asociado el usuario que se especifica por parámetros.

Entradas

<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrPerfilUsuario.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrPerfilUsuario.

Salida

<i>TrPerfilUsuario []</i>	Array de objetos <i>TrPerfilUsuario</i>
----------------------------	---

TrPerfilUsuario[] obtenerPerfilesUsuario(String usuario, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Entradas

<i>String usuario</i>	Identificador del usuario del que se quieren obtener sus perfiles. Si el identificador es 'null' o no válido se generará una excepción.
-----------------------	--

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrPerfilUsuario.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrPerfilUsuario.

Salida

TrPerfilUsuario [] Array de objetos *TrPerfilUsuario*

2.4.1.4.15.21 Provincias

TrProvincia[] obtenerProvincias(
String idProvincia,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos de la provincia que se indique por parámetro. Si se pasa 'null' se devolverán todas.

Entradas

String idProvincia Identificador de la provincia
Si el identificador es 'null' devolverá todos

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrProvincia.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrProvincia.

Salida

TrProvincia [] Array de objetos *TrProvincia*

2.4.1.4.15.22 Puestos de trabajo

TrPuestoTrabajo[] obtenerPuestosTrabajo(
String idPtoTrab,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos de los tipos de organismo que se indique en el parámetro *idTipoOrg*. Si se envía 'null' se devuelven todos.

Entradas

String idPtoTrab Identificador del puesto de trabajo
Si el identificador es 'null' devolverá todos

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrPuestoTrabajo.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrPuestoTrabajo.

Salida

TrPuestoTrabajo []

Array de objetos *TrPuestoTrabajo*

2.4.1.4.15.23 Puestos de trabajo por organismos

```
TrPtoTrabOrganismo[ ] obtenerPuestosTrabajoOrganismo(
    String idPtoTrab,
    TpoPK idOrganismo,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener los datos de los puestos de trabajo por organismos que se indiquen por parámetros. Si se pasan a 'null' se devolverán todos.

Entradas

<i>String idPtoTrab</i>	Identificador del puesto de trabajo Si el identificador es 'null' devolverá todos
<i>TpoPK idOrganismo</i>	Identificador del organismo Si el identificador es 'null' devolverá todos
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para <i>TrPtoTrabOrganismo</i> .
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para <i>TrPtoTrabOrganismo</i> .

Salida

TrPtoTrabOrganismo [] Array de objetos *TrPtoTrabOrganismo*

2.4.1.4.15.24 Sistemas

```
TrSistema[ ] obtenerSistemas(
    TpoPK idSistema,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener los datos del sistema que se indique en el parámetro *idSistema*. Si se envía 'null' se devuelven todos.

Entradas

<i>TpoPK idSistema</i>	Identificador del sistema Si el identificador es 'null' devolverá todos
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para <i>TrSistema</i> .
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para <i>TrSistema</i> .

Salida

TrSistema [] Array de objetos *TrSistema*

2.4.1.4.15.25 Obtener sistema establecido



TrSistema obtenerSistemaEstablecido() throws TrException;

Permite obtener el sistema establecido en la interfaz TrAPIUI.

Salidas

TrSistema

Sistema establecido en la interfaz TrAPIUI

2.4.1.4.15.26 Tipos de vía

TrTipoVia[] obtenerTiposVia(
String idTipoVia,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos de los tipos de vía que se indique en el parámetro *idTipoVia*. Si se pasan a 'null' se devolverán todos.

Entradas

String idTipoVia

Identificador del tipo de vía
Si el identificador es 'null' devolverá todos

ClausulaWhere where

Filtro a aplicar.
Consultar campos posibles para TrTipoVia.

ClausulaOrderBy orderBy

Ordenación a aplicar.
Consultar campos posibles para TrTipoVia.

Salida

TrTipoVia []

Array de objetos *TrTipoVia*

2.4.1.4.15.27 Usuarios

TrUsuario[] obtenerUsuarios(
String idUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos de los usuarios que se indique en el parámetro *idUsuario*. Si se pasan a 'null' se devolverán todos.

Entradas

String idUsuario

Identificador del usuario
Si el identificador es 'null' devolverá todos

ClausulaWhere where

Filtro a aplicar.
Consultar campos posibles para TrUsuario.

ClausulaOrderBy orderBy

Ordenación a aplicar.
Consultar campos posibles para TrUsuario.

Salida

TrUsuario []

Array de objetos *TrUsuario*

2.4.1.4.15.28 Obtener número de usuarios

```
int obtenerUsuariosCount(
    String idUsuario,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException
```

Devuelve el número de usuarios

Entradas

<i>String</i> idUsuario	Identificador del usuario Si el identificador es 'null' devolverá todos
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para <i>TrUsuario</i> .
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para <i>TrUsuario</i> .

Salida

<i>int</i>	Número de registros
------------	---------------------

2.4.1.4.15.29 Obtener tipo de componente

```
TrTipoComponente [ ] obtenerTipoComponente( TpoPK idTipoComp,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener uno o varios tipos de componentes. Si se pasa el parámetro *idTipoComp* se devuelve los datos de ese tipo de componente, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK</i> idTipoComp	Identificador del tipo de componente a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para <i>TrTipoComponente</i>
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para <i>TrTipoComponente</i>

Salida

<i>TrTipoComponente[]</i>	Array de objetos <i>TrTipoComponente</i>
----------------------------	--

2.4.1.4.15.30 Tipos de identificador

TrTipoidentificador[] obtenerTiposIdentificador(
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener los distintos tipos de identificador que existan.

Entradas

<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrTipoidentificador.
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrTipoidentificador.

Salida

TrTipoidentificador [] Array de objetos *TrTipoidentificador*

2.4.1.4.15.31 Tipos de organismos

TrTipoOrganismo[] obtenerTiposOrganismo(
 TpoPK idTipoOrg,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener los datos de los tipos de organismo que se indique en el parámetro *idTipoOrg*. Si se envía 'null' se devuelven todos.

Entradas

<i>TpoPK</i> idTipoOrg	Identificador del tipo de organismo Si el identificador es 'null' devolverá todos
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrTipoOrganismo.
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrTipoOrganismo.

Salida

TrTipoOrganismo [] Array de objetos *TrTipoOrganismo*

2.4.1.4.15.32 Número de tipos de expedientes

int obtenerTiposExpedienteCount(
 TpoPK idStma,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Devuelve el número de tipos de expedientes consultados.

Entradas

<i>TpoPK idStma</i>	Identificador del sistema. Si el identificador es 'null' el método devolverá todos los tipos de expediente definidos en TREW@.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrTipoExpediente. Los filtros que usen <i>TrTipoExpediente.CAMPO_REFSTMA</i> no tendrán efecto si el método recibe un identificador de sistema distinto de 'null'.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrTipoExpediente.

Salida

<i>Int]</i>	Número de registros
-------------	---------------------

2.4.1.4.15.33 Tipos de organización

TrTipoOrganizacion[] obtenerTiposOrganizacion(
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener los distintos tipos de organización que existan.

Entradas

<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrTipoOrganización.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrTipoOrganización.

Salida

<i>TrTipoOrganización []</i>	Array de objetos <i>TrTipoOrganización</i>
-------------------------------	--

2.4.1.4.15.34 Obtención de tipos pago del procedimiento

TrTipoPagoProcedimiento[] obtenerTiposPagoProcedimiento(TpoPK idDefProc,
 TpoPK tipoPago,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Devuelve el conjunto de tipo de pago del procedimiento indicado. Si no se indica procedimiento se devuelven todas.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador del procedimiento.</i>
<i>TpoPK tipoPago</i>	<i>Identificador del tipo de pago.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrTipoPagoProcedimiento.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrTipoPagoProcedimiento.

Salida

TrTipoPagoProcedimiento [] Array de objetos TrTipoPagoProcedimiento.

2.4.1.4.15.35 Obtener tipos de contacto

<p>TrTipoContacto[] obtenerTiposContacto(ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Permite obtener un conjunto de tipos de contactos.

Entradas

<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrTipoContacto.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrTipoContacto.

Salida

TrTipoContacto [] Array de objetos *TrTipoContacto*

2.4.1.4.15.36 Tipos de párrafos

TrTipoParrafo[] obtenerTiposParrafo(
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException;

Devuelve el conjunto de datos de los tipos de parrafos.

Entradas

<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrTipoParrafo.
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrTipoParrafo.

Salida

<i>TrTipoParrafo[]</i>	Array de objetos TrTipoParrafo con la información de tipos de expediente.
-------------------------	---

2.4.1.4.15.37 Tareas permitidas

TrTareaPermitida[] obtenerTareasPermitidas(
 TpoPK idFase,
 TpoPK idDefProc,
 TpoPK idExpediente,
 TpoPK idExpFase
 TpoDate fecha,
 String tareaLlamante,
 String condVisual,
 boolean soloUsuario,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Devuelve un conjunto de tareas permitidas que puede realizar un expediente desde una fase y un procedimiento dados. Mediante el parámetro *condVisual* indicamos qué tareas queremos obtener: "S": las que cumplen las condiciones de visualización, "N": las que no las cumplen y "T": todas. Con el parámetro *soloUsuario* se permite filtrar por el usuario establecido en el api y su perfil.

Entradas

<i>TpoPK idFase</i>	Identificador de la fase Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoPK idExpediente</i>	Identificador del expediente Opcional
<i>TpoPK idExpFase</i>	Identificador de la instancia de la fase. Si el identificador es 'null', el valor será la última instancia.
<i>TpoDate fecha</i>	Fecha Si es 'null' se coge la del sistema
<i>String tareaLlamante</i>	Tarea llamante



<i>String</i> condVisual	<i>Indica qué tareas se obtienen</i> → “ S ” – Las tareas que cumplen las condiciones de visualización → “ N ” – Las tareas que no cumplen las condiciones de visualización → “ T ” – Todas
<i>boolean</i> soloUsuario	<i>Indica si se filtra por el usuario establecido en el api y su perfil</i>
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrTareaPermitida.
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrTareaPermitida.

Salida

TrTareaPermitida [] *Array de objetos TrTareaPermitida.*

TrTareaPermitida[] obtenerTareasPermitidas(

TpoPK idFase,
TpoPK idDefProc,
TpoPK idExpediente,
TpoDate fecha,
String tareaLlamante,
String condVisual,
boolean soloUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Entradas

<i>TpoPK idFase</i>	<i>Identificador de la fase</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoPK idExpediente</i>	<i>Identificador del expediente</i> Opcional
<i>TpoDate fecha</i>	<i>Fecha</i> Si es 'null' se coge la del sistema
<i>String tareaLlamante</i>	<i>Tarea llamante</i>
<i>String condVisual</i>	<i>Indica qué tareas se obtienen</i> → “ S ” – Las tareas que cumplen las condiciones de visualización → “ N ” – Las tareas que no cumplen las condiciones de visualización → “ T ” – Todas
<i>boolean soloUsuario</i>	<i>Indica si se filtra por el usuario establecido en el api y su perfil</i>
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrTareaPermitida.
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrTareaPermitida.

Salida

TrTareaPermitida [] *Array de objetos TrTareaPermitida.*

2.4.1.4.15.38 Obtener número de sistemas

int obtenerSistemasCount(
 TpoPK idSistema,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Entradas

<i>TpoPK idSistema</i>	<i>Identificador del sistema</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrTareaPermitida.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrTareaPermitida.</i>

Salida

TrTareaPermitida [] *Array de objetos TrTareaPermitida.*

2.4.1.4.15.39 Obtención de los tipos de actos de un expediente

TrTipoActo[] obtenerTiposActoAdmExpediente(
 TpoPK idExpediente,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite al usuario obtener los datos del conjunto de tipos de acto administrativos que están representados mediante transiciones en un expediente concreto.

Entradas

<i>TpoPK idExpediente</i>	<i>Identificador de la definición del procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrTipoActo.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrTipoActo.</i>

Salida

TrTipoActo [] *Array de objetos TrTipoActo con la información referente a los tipos de actos administrativos que están representados mediante transiciones.*

2.4.1.4.15.40 Tipos de documento

String[] obtenerTiposDocumentoPortafirmas() throws TrException

Método que devuelve la lista de tipos de documentos dados de alta en Port@firmas

Salida

String[] Array de objetos String's con los tipos de documentos admitidos

2.4.1.4.15.41 Tipos de indicación

TrTipoIndicacion[] obtenerTipoIndicacion(TpoPK idTipoInd, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener un conjunto de tipos de indicaciones definidas en Trew@.

Entradas

<i>TpoPK idTipoInd</i>	Identificador del tipo de indicación. Si no se indica se devuelven todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrTipoIndicacion.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrTipoIndicacion.

Salida

TrTipoIndicacion [] Array de objetos TrTipoIndicacion

2.4.1.4.15.42 Relaciones establecidas

TrRelacionDefinida[] obtenerRelacionesDefinidas(TpoPK idStma, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Mediante este procedimiento se obtienen las relaciones establecidas entre fases, transiciones, tipos de evolución, etc., definidas en el tramitador para un sistema que se indica como parámetro.

Entradas

<i>TpoPK idStma</i>	Identificador del sistema. Si el identificador es 'null' el método devolverá todas las relaciones definidas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrRelacionDefinida. Los filtros que usen TrRelacionDefinida CAMPO_REFSTMA no tendrán efecto si el método recibe un identificador de sistema distinto de 'null'.
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrRelacionDefinida.

Salida

TrRelacionDefinida [] Array de objetos *TrRelacionDefinida*

2.4.1.4.15.43 Obtener valor metadato

Object obtenerValorMetadato(
TpoPK idMetadato,
TpoPK idExp,
TpoPK idDoc,
TpoPK idTrans,
TpoPK idDocPer,
TpoPK idExpEnFase,
TpoPK idDefProc,
Timestamp fecha,
String usuario,
TpoPK idFase,
TpoPK idTipoDoc) throws *TrException*

Obtiene el valor del metadato mediante la llamada por reflexión al paquete y método definido en la instancia del metadato.

Entradas

<i>TpoPK idMetadato</i>	Identificador del metadato del que se va a obtener el valor.
<i>TpoPK idExp</i>	Identificador del expediente al que está asociado el metadato.
<i>TpoPK idDoc</i>	Identificador del documento al que está asociado el metadato.
<i>TpoPK idTrans</i>	Identificador de la transición
<i>TpoPK idDocPer</i>	Identificador del documento permitido.
<i>TpoPK idExpEnFase</i>	Identificador del expediente en fase.
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento.
<i>Timestamp fecha</i>	Fecha.
<i>String usuario</i>	Usuario.

TpoPK idFase Identificador de la fase.

TpoPK idTipoDoc Identificador del tipo de documento.

Salida

Object Resultado del método por reflexión

2.4.1.4.15.44 Variables sistema

TrVariable[] obtenerVariablesSistema(
TpoPK idStma,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws *TrException*

Permite obtener los valores de las variables de un sistema.

Entradas

TpoPK idStma Identificador del sistema.
Si el identificador es 'null' el método devolverá todas las variables del sistema.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrVariable.

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrVariable.

Salida

TrVariable [] Array de objetos TrVariable con la información de las variables del sistema

2.4.1.4.15.45 Devuelve el valor de la variable.

String obtenerValorVariable(
TpoPK idVariable,
TpoPK idDocExp) throws *TrException* *Deprecado*

Devuelve el valor de la variable utilizando los parámetros asignados al documento.

Entradas

<i>TpoPK idVariable</i>	Identificador de la variable
<i>TpoPK idDocExp</i>	Identificador del documento

Salida

<i>String</i>	El valor de la variable
---------------	-------------------------

2.4.1.4.15.46 Obtener URL del documento generado

```
String obtenerURLDocumentoGenerado(  
    TpoPK idDocExp,  
    String multiple) throws TrException
```

Devuelve la URL del documento generado. Sólo válido para documentos de report.

Entrada

<i>TpoPK idDocExp</i>	Identificador del documento.
<i>String multiple</i>	Indica si el documento es múltiple o no.

Salida

<i>String</i>	URL del documento generado.
---------------	-----------------------------

2.4.1.4.15.47 Obtención de procedimientos y versiones de procedimientos

```
TrVersionDefProcedimiento[ ] obtenerVersionesDefProcedimiento(  
    TpoPK idTipoExp,  
    ClausulaWhere where,  
    ClausulaOrderBy orderBy ) throws TrException
```

Devuelve para un tipo de expediente dado, los datos de todas las versiones de procedimiento que tiene asociados. Si no se indica tipo de expediente, devuelve todas las versiones definidas en TREW@.

Entradas

<i>TpoPK idTipoExp</i>	<i>Identificador del tipo de expediente.</i> Si el identificador es 'null', el método devolverá todas las versiones definidas en TREW@.
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrVersionDefProcedimiento. Los filtros que usen <i>TrVersionDefProcedimiento.CAMPO_REFTIPOEXP</i> no tendrán efecto si el método recibe un identificador del tipo de expediente distinto de 'null'.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrVersionDefProcedimiento.

Salida

TrVersionDefProcedimiento [] Array de objetos TrVersionDefProcedimiento.

2.4.1.4.15.48 Obtener variables del documento del expediente

```
TrVariableDocExp[] obtenerVariablesDocumentoExp(
    TpoPK idDocExp,
    ClausulaWhere where,
    ClausulaOrderBy orderBy) throws TrException
```

Devuelve el conjunto de variables asociadas al documento del expediente que se pase como parámetro.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento del expediente.</i> Si el identificador es 'null', o no válido, se generará una excepción
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> Consultar campos posibles para TrVariableDocExp.
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> Consultar campos posibles para TrVariableDocExp.

Salida

TrVariableDocExp[] Array de objetos TrVariableDocExp

2.4.1.4.15.49 Reanudar tarea del expediente

```
void reanudarTareaExpediente(
    TpoPK idTareaExp,
    String tipoTarea) throws TrException
```

Permite reanudar una tarea de un expediente. El tipo de la tarea puede ser: **"I"**: Para un documento incorporado; **"G"**: Para un documento generado; **"M"**: Tarea del tipo manipular datos; **"O"**: Otros tipos de tarea.

Entradas

<i>TpoPK</i> idTareaExp	Identificador de la tarea Si el identificador es 'null', o no válido, se generará una excepción
<i>String</i> tipoTarea	Tipo de tarea → "I" – Incorporar → "G" – Generar → "M" – Manipular datos → "O" – Otras

2.4.1.4.15.50 Recuperar contenido de documento externo

InputStream recuperarDocExternoExpediente(
TpoPK idDocExpte,
TpoString formato,
TpoString nombreFichero
String motivo) throws *TrException*

Permite descargar el contenido del documento externo. Los parámetros *formato* y *nombreFichero* son de Entrada / Salida por si se quiere obtener el valor de dichos atributos.

Entradas

<i>TpoPK</i> idDocExpte	Identificador del documento. Si el identificador es 'null', o no válido, se generará una excepción
<i>TpoString</i> formato	Tipo "mime" del fichero. Entrada/Salida.
<i>TpoString</i> nombreFichero	Nombre del fichero. Entrada/Salida.
<i>String</i> motivo	Motivo para realizar la consulta al repositorio externo.

Salida

<i>InputStream</i>	Fichero físico asociado al documento.
--------------------	---------------------------------------

2.4.1.4.15.51 Obtener valores metadatos

Map<String, String> obtenerValoresMetadatos(
List<*TpoPK*> idMetadatos,
TpoPK idExp,

```
TpoPK idDoc,
TpoPK idTrans,
TpoPK idDocPer,
TpoPK idExpEnFase,
TpoPK idDefProc,
Timestamp fecha,
String usuario,
TpoPK idFase,
TpoPK idTipoDoc) throws TrException
```

Obtiene el valor del metadato mediante la llamada por reflexión al paquete y método definido en la instancia del metadato.

Entradas

<i>List</i> < <i>TpoPK</i> > idMetadato	Lista de identificadores de metadatos a consultar
<i>TpoPK</i> idExp	Identificador del expediente al que está asociado el metadato.
<i>TpoPK</i> idDoc	Identificador del documento al que está asociado el metadato.
<i>TpoPK</i> idTrans	Identificador de la transición
<i>TpoPK</i> idDocPer	Identificador del documento permitido.
<i>TpoPK</i> idExpEnFase	Identificador del expediente en fase.
<i>TpoPK</i> idDefProc	Identificador de la definición del procedimiento.
<i>Timestamp</i> fecha	Fecha.
<i>String</i> usuario	Usuario.
<i>TpoPK</i> idFase	Identificador de la fase.
<i>TpoPK</i> idTipoDoc	Identificador del tipo de documento.

Salida

<i>Object</i>	Resultado del método por reflexión
---------------	------------------------------------

2.4.1.4.15.52 Filtro de firmantes

```
void setFiltroFirmantes(boolean filtroFirmantes) throws TrException
```

Permite indicar si la obtención de firmantes de un tipo de documento se filtrará por el tipo de Unidad Orgánica (organismo) y por provincia. Por defecto su valor es true, con lo cuál se realizará el filtro para mantener compatibilidad con versiones anteriores. Si se indica el parámetro filtroFirmantes a false el filtro no se aplicará y se devolverán todos los firmantes que haya definido para el tipo de documento.

Entradas

<i>boolean</i> filtroFirmantes	<i>Indica si se aplica el filtro de firmantes</i> Por defecto su valor es true.
--------------------------------	--

2.4.1.4.15.53 Suspender o reanudar el plazo para una caducidad del expediente

```
void suspenderReanudarCaducidad(  
    TpoPK idCaducidadExp,  
    String tipo,  
    TpoDate fecha) throws TrException
```

Permite modificar una caducidad del expediente, suspendiendo o reanudando el plazo.

Entradas

<i>TpoPK</i> idCaducidadExp	<i>Identificador de la caducidad del expediente.</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>String</i> tipo	<i>Tipo de la modificación.</i> → “ S ” – Suspender el plazo → “ R ” – Reanudar el plazo
<i>TpoDate</i> fecha	<i>Fecha en la que se produce la suspensión op reanudación.</i> <i>Si no se indica, toma la fecha del sistema.</i>

2.4.1.4.15.54 Siguiente fecha hábil

```
Timestamp siguienteFechaHabil(  
    TpoPK idExpediente,  
    Timestamp fechaInicial,  
    String unidad,  
    int numUnidades ) throws TrException
```


Devuelve la siguiente fecha hábil en función del expediente indicado, la fecha indicada, la unidad de tiempo y el número de unidades. Los días festivos se obtienen a partir de la provincia y municipio del organismo al que pertenece el expediente.

Entradas

<i>TpoPK</i> idExpediente	<i>Identificador del expediente</i> <i>Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>Timestamp</i> fechalnicial	<i>Fecha inicial</i>
<i>String</i> unidad	<i>Unidad de tiempo.</i> Sus valores pueden ser TrDiaFestivo.DIA, TrDiaFestivo.MES ó TrDiaFestivo.ANYO.
<i>int</i> numUnidades	<i>Número de unidades (días, meses o años)</i>

Salida

<i>Timesamtp</i>	<i>Siguiente fecha hábil</i>
------------------	------------------------------

```
Timestamp siguienteFechaHabil(
    String codProvincia
    String codMunicipio
    Timestamp fechalnicial,
    String unidad,
    int numUnidades ) throws TrException
```

Devuelve la siguiente fecha hábil en función de la provincia y el municipio indicados, la fecha indicada, la unidad de tiempo y el número de unidades. Los días festivos se obtienen a partir de la provincia y municipio del organismo al que pertenece el expediente.

Entradas

<i>String</i> codProvincia	<i>Identificador de la provincia.</i> Si se indica null se calculará con los festivos nacionales.
<i>String</i> codMunicipio	<i>Identificador del municipio.</i> Si se indica el municipio la provincia es obligatoria. Si no se indica se calculará con los festivos provinciales (si se ha indicado provincia) ó nacionales si no se ha indicado ninguno.
<i>Timestamp</i> fechalnicial	<i>Fecha inicial</i>
<i>String</i> unidad	<i>Unidad de tiempo.</i> Sus valores pueden ser TrDiaFestivo.DIA, TrDiaFestivo.MES ó TrDiaFestivo.ANYO.
<i>int</i> numUnidades	<i>Número de unidades (días, meses o años)</i>

Salida

<i>Timesamtp</i>	<i>Siguiente fecha hábil</i>
------------------	------------------------------

2.4.1.4.15.55 Solicitar alta abonado

```
void solicitarAltaAbonadoNotifica (TrInteresadoExpediente interesadoExp,
    String firmaServicio, String formReference,
    String transactionID) throws TrException;
```

Método para solicitar el alta de un interesado como abonado en un servicio de notificaciones.

Entrada

<i>TrInteresadoExpediente</i> interesadoExp	Interesado del expediente que solicita el alta en el servicio
<i>String</i> firmaServicio	Firma de la autorización del alta al servicio.
<i>String</i> formReference	Identificador del formulario firmado
<i>String</i> transactionID	Identificador de la transacción de firma

void solicitarAltaAbonadoNotifica (TrInteresadoExpediente interesadoExp) throws TrException;

Entrada

<i>TrInteresadoExpediente</i> interesadoExp	Interesado del expediente que solicita el alta en el servicio
--	---

2.4.1.4.15.56 Revisar notificaciones pendientes

void revisaNotificacionesPendientes(TpoPK idExpediente,
TpoPK idDocExp,
TpoPK idInteresado) throws TrException;

Método que revisa el estado de las notificaciones que estén pendientes, para el expediente, documento e interesado indicados.

Entrada

<i>TpoPK</i> idExpediente	Identificador del expediente. Opcional.
<i>TpoPK</i> idDocExp	Identificador del documento del expediente. Opcional.
<i>TpoPK</i> idInteresado	Identificador del interesado. Opcional.

void revisaNotificacionesPendientes() throws TrException;

2.4.1.4.15.57 Revocar delegar sustituir firma

void revocarDelegacionSustitucionFirmante(
String idPtoTrab,
TpoPK idUniOrg,

```
String idPtoTrabSust,
TpoPK idUniOrgSust,
TpoDate fecha,
String tipo,
TpoPK idTipoDoc) throws TrException
```

Complementaria a la anterior, permite revocar las delegaciones/sustituciones de firma.

Entradas

<i>String idPtoTrab</i>	<i>Identificador del puesto de trabajo del firmante principal.</i>
<i>TpoPK idUniOrg</i>	<i>Identificador de la unidad orgánica del firmante principal</i>
<i>String idPtoTrabSust</i>	<i>Identificador del puesto de trabajo del delegado/sustituto</i>
<i>TpoPK idUniOrgSust</i>	<i>Identificador de la Unidad orgánica del delegado/sustituto</i>
<i>TpoDate fecha</i>	<i>Fecha de revocación Si no se indica nada, por defecto toma la del sistema.</i>
<i>String tipo</i>	<i>Tipo → "D" – Delegación (valor por defecto) → "S" – Sustitución</i>
<i>TpoPK idTipoDoc</i>	<i>Identificador del tipo de documento en el que se delega (para el caso de delegaciones).</i>

2.4.1.4.15.58 Obtención de usuarios asignados a un expediente

```
TrUsuarioAsignado[ ] obtenerUsuariosAsignadosExpediente(  
TpoPK idExpediente,  
ClausulaWhere where,  
ClausulaOrderBy orderBy) throws TrException
```

Devuelve los usuarios asignados al expediente y datos de la propia asignación.

Entradas

<i>TpoPK IdExpediente</i>	<i>Identificador del expediente. Si el identificador es 'null' se obtendrán los usuarios asignados a todos los expedientes.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para <i>TrUsuarioAsignado</i></i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para <i>TrUsuarioAsignado</i>.</i>

Salida

TrUsuarioAsignado[] *Array de objetos TrUsuarioAsignado*

2.4.1.4.15.59 Obtención de los tipos de actos de un procedimiento

```
TrTipoActo[ ] obtenerTiposActoAdmDefProcedimiento(  
TpoPK idDefProc,  
ClausulaWhere where,  
ClausulaOrderBy orderBy ) throws TrException
```

Permite al usuario obtener los datos del conjunto de tipos de acto administrativos que están representados mediante transiciones en un procedimiento concreto.

Entradas

<i>TpoPK idDefProc</i>	<i>Identificador de la definición del procedimiento. Si el identificador es 'null', o no válido, se generará una excepción.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar. Consultar campos posibles para TrTipoActo.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar. Consultar campos posibles para TrTipoActo.</i>

Salida

<i>TrTipoActo []</i>	<i>Array de objetos TrTipoActo con la información referente a los tipos de actos administrativos que están representados mediante transiciones.</i>
-----------------------	---

2.4.1.4.15.60 (w) Versionar documento

TpoPK versionarDocumentoExpediente(
TpoPK idDocExp,
TpoDate fechaFin) throws *TrException*

Permite versionar un documento del expediente creando una copia exacta del documento al que versiona con todos sus párrafos, variables, parámetros, firmas, interesados en el documento, notificaciones.

El documento al que versiona se le cambia el estado a "V" (Versionado) y se actualiza su fecha de finalización, y al nuevo documento se le asigna un número de versión posterior al que versiona.

Si se ha definido un componente w@rdA en el sistema establecido en el TrAPIUI se creará el nuevo documento versionado. Si el documento al que versiona estaba guardado en el mismo componente w@rdA que está definido en el sistema se actualizará su estado.

Entradas

<i>TpoPK idDocExp</i>	<i>Identificador del documento del expediente. Si el identificador es 'null', o no válido, se generará una excepción</i>
<i>TpoDate fechaFin</i>	<i>Fecha de finalización del documento que se versiona Si es 'null' por defecto se coge la del sistema</i>

Salida

<i>TpoPK</i>	<i>Identificador del nuevo documento del expediente</i>
--------------	---

2.4.1.4.15.61 Commit y Rollback

boolean commit()

Realiza un commit sobre la base de datos guardando los últimos cambios.

Salida

boolean

→ **"true"** – El commit se ha realizado correctamente
→ **"false"** – El commit no se ha realizado correctamente

boolean rollback()

Realiza un rollback sobre la base de datos deshaciendo los últimos cambios.

Salida

boolean

→ **"true"** – El rollback se ha realizado correctamente
→ **"false"** – El rollback no se ha realizado correctamente

2.4.1.4.15.62 AutoCommit

void setAutoCommit(boolean value)

Establece el valor del autoCommit.

Entrada

boolean value

→ **"true"** – Se realiza un commit cada vez que se ejecute un api.
→ **"false"** – No se realiza commit cuando se ejecute un api, el usuario debe hacer uso del método commit() para guardar los cambios.

boolean getAutoCommit()

Devuelve el estado actual del autoCommit.

Salida

boolean

→ **"true"** – Se realiza un commit cada vez que se ejecute un api.
→ **"false"** – No se realiza commit cuando se ejecute un api, el usuario debe hacer uso del método commit() para guardar los cambios.

2.4.1.4.15.63 Obtener usuario establecido

String obtenerUsuarioSistema(TpoString rol)

Permite obtener el usuario establecido en el TrAPIUI y su rol.

Entradas/Salidas

TpoString rol

*Rol del usuario establecido en el api
Parámetro de entrada/salida*

Salida

String usuario

Usuario establecido en el api

String obtenerUsuarioSistema()

2.4.1.4.15.64 Establecer conexión fija

synchronized void establecerConexionFija(boolean fija)
void setConexionFija(boolean fija)

Permite establecer la conexión como fija, es decir una vez que se crea no se cierra hasta que no se ejecute el método cerrarSesion. Este método sólo tiene efecto cuando la TrAPIUI se ha creado con un perfil de conexión que sea un datasource o se use properties con pool de conexiones.

Si se está usando un perfil de conexión que es un datasource o un properties con el pool activado, y el conexionFija es false se devuelve la conexión al finalizar la ejecución de un método, en caso contrario no se devuelve.

Entrada

boolean fija

Indica si la conexión es fija para la TrAPIUI o se devuelve al pool de conexiones al finalizar la ejecución de un método.

→ **“true”** – La conexión es fija y no se devuelve al pool. Por defecto toma este valor.

→ **“false”** – Se devuelve la conexión al pool al terminar la ejecución de cada método, haciéndose commit al final de cada método.

2.4.1.4.15.65 Obtener estado conexión fija

synchronized boolean obtenerEstadoConexionFija()



boolean isConexionFija()

Devuelve el valor actual del atributo `conexionFija`. El estado de la conexión fija sólo tiene efecto cuando la TrAPIUI se ha creado usando un perfil de conexión que es un `datasource`. Si el valor devuelto es `true` indica que la conexión se fija y no se devuelve al pool hasta cerrar la sesión, en caso contrario si el valor es `false` indica que no es fija y que se devuelve la conexión al pool al finalizar la ejecución de cada método.

Salida

boolean

Valor del atributo `conexionFija`

→ **“true”** – La conexión es fija y no se devuelve al pool. Por defecto toma este valor.

→ **“false”** – Se devuelve la conexión al pool al terminar la ejecución de cada método, haciéndose `commit` al final de cada método.

2.4.1.4.15.66 Cierre de la sesión

void cerrarSesion()

Cierra la sesión actual de trabajo, haciendo por defecto un `commit` si el `autoCommit` está a `true`, si no se realiza un `rollback`.

void cerrarSesion(boolean commit)

Cierra la sesión actual de trabajo. Realiza un `commit` o un `rollback` según se indique en el parámetro `commit`.

Entrada

boolean commit

→ **“true”** – Se realiza un `commit` para guardar los cambios.

→ **“false”** – Se realiza un `rollback` para no guardar los cambios, desde el último `commit`.

2.4.1.4.15.67 Comprobación de conexión

boolean hayConexion()

Indica si existe o no una conexión válida a la base de datos.

Salida

boolean

→ **“true”** – Existe una conexión válida a la base de datos.

→ **“false”** – No existe conexión válida a la base de datos.

2.4.1.4.15.68 Establecer el usuario

```
void establecerUsuarioSistema(  
    String usuario) throws TrException
```

Permite establecer el usuario.

Entradas

String usuario

Usuario

Los usuarios deben tener el rol o perfil de usuario TR_R_USUARIO o el TR_R_ADMINISTRADOR.

2.4.1.4.15.69 Comprobar bloqueo

```
void setComprobarBloqueo(  
    Boolean bloq) throws TrException
```

Establece el valor del comprobarBloqueo.

Entrada

boolean value

→ **"true"** – Se comprueba si el expediente se encuentra bloqueado
→ **"false"** – No se comprueba si el expediente se encuentra bloqueado

```
boolean getComprobarBloqueo()
```

Devuelve el estado actual del comprobarBloqueo.

Salida

boolean

→ **"true"** – Se comprueba si el expediente se encuentra bloqueado
→ **"false"** – No se comprueba si el expediente se encuentra bloqueado

En caso de no establecer ninguna preferencia el valor de este atributo será **true**

2.4.2 Descripción de la TrAPIADM

2.4.2.1 Descripción de clases involucradas

2.4.2.1.1 Interfaz J-TrAPIADM

La interfaz J-TrAPIADM define todos los métodos de administración del motor de tramitación TREW@. En la librería este acceso se realiza a través de la implementación de la interfaz **TrAPIADM** que se describe en este apartado.

2.4.2.1.1.1 TrAPIADM

trewa.bd.trapi.trapiadm.TrAPIADM

Interfaz de administración del motor de tramitación TREW@. Todos los métodos quedan descritos en el apartado “**Métodos del TrAPIADM**”.

Para poder trabajar con la TrAPIADM es preciso crear una instancia de la misma a través de la clase Factoría: **TrAPIADMFactory**, descrita en el apartado siguiente.

2.4.2.1.2 Clase Factoría

La clase Factoría se encarga de crear una instancia correcta de la J-TrAPIADM de forma que los métodos de esta última aparezcan disponibles para las aplicaciones.

2.4.2.1.2.1 TrAPIADMFactory

trewa.bd.trapi.trapiadm.TrAPIADMFactory

Clase que permite la correcta creación de una instancia de la TrAPIADM.

Métodos para la creación de instancias del TrAPIADM

→ TrAPIADM crearAPIADM ()

Devuelve una instancia del TrAPIADM. La información para el establecimiento de la conexión la obtiene a través del perfil “default”.

Salidas *TrAPIADM* Instancia del TrAPIADM o “null” si no se pudo crear.

→ TrAPIADM crearAPIADM(String usuario, String clave)

Devuelve una instancia del TrAPIADM. La información para el establecimiento de la conexión la obtiene a través del perfil “default” pero en este caso hace uso del usuario y password que recibe como parámetros.

Entradas

String usuario	Usuario
String clave	Clave de usuario

Salidas *TrAPIADM* Instancia del TrAPIADM o “null” si no se pudo crear.

→ TrAPIADM crearAPIADM(String strPerfil)

Devuelve una instancia del TrAPIADM. La información para el establecimiento de la conexión la obtiene a través del perfil especificado en la llamada como parámetro.

Entradas *String strPerfil* *Nombre del perfil para el establecimiento de la conexión.*

Salidas *TrAPIADM* *Instancia del TrAPIADM o “null” si no se pudo crear.*

→ TrAPIADM crearAPIADM(String strPerfil, String usuario, String clave)

Devuelve una instancia del TrAPIADM. La información para el establecimiento de la conexión la obtiene a través del perfil especificado en la llamada como parámetro, pero en este caso hace uso del usuario y password que recibe como parámetros.

Entradas *String strPerfil* *Nombre del perfil para el establecimiento de la conexión.*

String usuario Usuario

String clave Clave de usuario

Salida *TrAPIADM* *Instancia del TrAPIADM o “null” si no se pudo crear.*

Ejemplo de uso

En el siguiente ejemplo se muestra cómo obtener una instancia del TrAPIADM a través de la clase TrAPIADMFactory:

```
TrAPIADM apiADM = null;

apiADM = TrAPIADMFactory.crearAPIADM( "perfilTrewa");
if ( apiADM != null ){
    // Ya se puede acceder a los métodos del TrAPIADM
    pK = new TpoPK( BigDecimal.valueOf(39) );
    try{
        TrDefProcedimiento[] def = apiADM.obtenerDefProcedimiento(null,null,null);
        ...
    }
}
catch(TrException ex)
{
    ex.printStackTrace();
}
}
```

2.4.2.1.3 Clases de entidad

Estas clases son el mecanismo de intercambio de información entre las aplicaciones y el motor Trew@ a través de las J-TrAPIs.

Las clases de entidad asociadas al TrAPIADM se encuentran en el paquete **trewa.bd.trapi.tpo**. Estas clases de entidad permiten realizar todo el mantenimiento del motor de tramitación Trew@ mediante el uso de métodos de inserción, modificación, eliminación y obtención.

2.4.2.1.3.1 TrAccion

trewa.bd.trapi.tpo.TrAccion

Clase que representa la información de una acción.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFACCION	TpoPK	Identificador de la acción.
→ NOMBRE	String	Nombre con el que se denomina la acción.
→ DESCRIPCION	String	Texto que describe la acción.
→ PARAMREFEXP	String	Indica si se le pasa el id del expediente a la acción. → "S" – Sí → "N" – No
→ PARAMREFTRAN	String	Indica si se le pasa el id de la transición a la acción. → "S" – Sí → "N" – No
→ PARAMREFDOCPER	String	Indica si se le pasa el id del documento permitido a la acción. → "S" – Sí → "N" – No
→ PARAMREFEXPXNAS	String	Indica si se le pasa el id del paso en la evolución actual a la acción. → "S" – Sí → "N" – No
→ PARAMREFDEFPROC	String	Indica si se le pasa el id de la definición del procedimiento a la acción. → "S" – Sí → "N" – No
→ PARAMFECHA	String	Indica si se le pasa la fecha de tramitación/generación/incorporación indicada a la acción. → "S" – Sí → "N" – No
→ PARAMUSUARIO	String	Indica si se le pasa el usuario a la acción. → "S" – Sí → "N" – No
→ PARAMREFFASE	String	Indica si se le pasa el identificador de la fase a la acción. → "S" – Sí → "N" – No
→ PARAMREFTIPODOC	String	Indica si se le pasa el identificador del tipo de documento a la acción. → "S" – Sí → "N" – No

→ IMPLEMENTACION	String	Indica el tipo de implementación asociada a la acción. → “ F ” – Función en el servidor → “ J ” – Java
→ PAQUETE	String	Indica el paquete que contiene la función a ejecutar si se trata de una función en el servidor, o el nombre del paquete y clase al que pertenece la función java a ejecutar.
→ NOMBFUNCION	String	Indica el nombre de la función a ejecutar.
→ STMA	TrSistema	Sistema de la acción.
→ TEXTO	String	Texto libre a pasar por defecto a la acción

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFACCION	NUMÉRICO	Identificador de la acción
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la acción
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la acción
→ CAMPO_PARAMREFEX P	CADENA(1)	Indica si se le pasa el id del expediente a la acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFTR AN	CADENA(1)	Indica si se le pasa el id de la transición a la acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFDO CPER	CADENA(1)	Indica si se le pasa el id del documento permitido a la acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFEX PXFAS	CADENA(1)	Indica si se le pasa el id del paso en la evolución actual a la acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFDE FPROC	CADENA(1)	Indica si se le pasa el id de la definición del procedimiento a la acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMFECHA	CADENA(1)	Indica si se le pasa la fecha de tramitación/generación/incorporación indicada a la acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMUSUAR IO	CADENA(1)	Indica si se le pasa el usuario a la acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFFA SE	CADENA(1)	Indica si se le pasa el identificador de la fase a la acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFTIP ODOC	CADENA(1)	Indica si se le pasa el identificador del tipo de documento a la acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_IMPLEMENTA CION	CADENA(1)	Indica el tipo de implementación asociada a la acción. → “ F ” – Función en el servidor → “ J ” – Java
→ CAMPO_PAQUETE	CADENA(100)	Indica el paquete que contiene la función a ejecutar si se trata de una función en el servidor, o el nombre del paquete y clase al que

- **CAMPO_NOMBFUNCION** CADENA(50) pertenece la función java a ejecutar.
Indica el nombre de la función a ejecutar
- **CAMPO_REFSTMA** NUMÉRICO Identificador del sistema
- **CAMPO_TEXTO** CADENA(50) Texto libre a pasar por defecto a la acción

Métodos para aplicación de los filtros → obtenerAccion

2.4.2.1.3.2BloquePermitido

trewa.bd.trapi.tpo.TrAccionBloquePermitido

Representa la información de una acción asociada a una tarea en fase.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ ACCION	TrAccion	Acción
→ BLOQUEPER	TrBloquePermitido	Bloque permitido.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ MENSAJEOK	String	Mensaje a mostrar cuando se cumple la acción.
→ MENSAJENOOK	String	Mensaje a mostrar cuando la acción no se cumple.
→ VALIDA	String	Indica si la acción asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ MOSTRARMSJ	String	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ COMPROBAR	String	Indica si se realiza la acción. → “ I ” – La acción se produce al Incorporar → “ V ” – La acción se produce al Visualizar
→ TEXTO	String	Texto libre a pasar a la acción asociada al bloque permitido

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFACCION	NUMÉRICO	Identificador de la acción
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→ CAMPO_REFBLOQUEPER	NUMÉRICO	Identificador del bloque permitido o tarea en fase.
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase
→ CAMPO_REFBLOQUEIN	NUMÉRICO	Identificador del bloque inicial o tarea llamante

I		
→ CAMPO_REFBLOQUEFI	NUMÉRICO	Identificador del bloque final o tarea
N		
→ CAMPO_VALIDA	CADENA(1)	Indica si la acción asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza la acción. → “ I ” – La acción se produce al Incorporar → “ V ” – La acción se produce al Visualizar
→ CAMPO_MOSTRARMSJ	CADENA(1)	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar cuando la acción se cumple.
→ CAMPO_MENSAJENOO	CADENA(200)	Mensaje a mostrar cuando la acción no se cumple.
K		
→ CAMPO_TEXTO	CADENA(50)	Texto libre a pasar a la acción asociada al bloque permitido

Métodos para aplicación de los filtros

→ obtenerAccionBloquePermitido

2.4.2.1.3.3 TrAccionDocumentoPermitido

trewa.bd.trapi.tpo.TrAccionDocumentoPermitido

Clase que representa la información de una acción asociada a un documento permitido.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ ACCION	TrAccion	Acción
→ TIPODOC	TrTipoDocumento	Tipo de documento.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ FASE	TrFase	Fase.
→ MENSAJEOK	String	Mensaje a mostrar cuando se cumple la acción.
→ MENSAJENOOK	String	Mensaje a mostrar cuando la acción no se cumple.
→ VALIDA	String	Indica si la acción asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ MOSTRARMSJ	String	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ COMPROBAR	String	Indica si se realiza la acción. → “ G ” – La acción se produce al Generar el documento → “ I ” – La acción se produce al Incorporar el documento → “ T ” – La acción se produce al Generar e Incorporar el



→ **TEXTO** String documento
→ “**V**” – La acción se produce al Visualizar el documento
Texto libre a pasar a la acción asociada al documento permitido

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFACCION	NUMÉRICO	Identificador de la acción
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar cuando se cumple la acción.
→ CAMPO_MENSAJENOOK	CADENA(200)	Mensaje a mostrar cuando no se cumple la acción.
→ CAMPO_VALIDA	CADENA(1)	Indica si la acción asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ CAMPO_MOSTRARMSJ	CADENA(1)	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza la acción. → “ G ” – La acción se produce al Generar el documento → “ I ” – La acción se produce al Incorporar el documento → “ T ” – La acción se produce al Generar e Incorporar el documento → “ V ” – La acción se produce al Visualizar el documento
→ CAMPO_TEXTO	CADENA(50)	Texto libre a pasar a la acción asociada al documento permitido

Métodos para aplicación de los filtros

→ obtenerAccionDocumentoPermitido

2.4.2.1.3.4TrAccionTransicion

trewa.bd.trapi.tpo.TrAccionTransicion

Clase que representa la información de una acción asociada a una transición.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ ACCION	TrAccion	Acción
→ TRANSICION	TrTransicion	Transición.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ MENSAJEOK	String	Mensaje a mostrar cuando se cumple la acción.

→ MENSAJENOOK	String	Mensaje a mostrar cuando la acción no se cumple.
→ VALIDA	String	Indica si la acción asociada a la transición es válida. → “ S ” – Sí → “ N ” – No
→ MOSTRARMSJ	String	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ COMPROBAR	String	Indica si se realiza la acción. → “ T ” – La acción se produce al Tramitar → “ D ” – La acción se produce al Deshacer → “ A ” – La acción se produce al Tramitar y Deshacer → “ V ” – La acción se produce al Visualizar
→ TEXTO	String	Texto libre a pasar a la acción asociada a la transición

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFACCION	NUMÉRICO	Identificador de la acción
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_VALIDA	CADENA(1)	Indica si la acción asociada a la transición es válida. → “ S ” – Sí → “ N ” – No
→ CAMPO_MOSTRARMSJ	CADENA(1)	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza la acción. → “ T ” – La acción se produce al Tramitar → “ D ” – La acción se produce al Deshacer → “ A ” – La acción se produce al Tramitar y Deshacer → “ V ” – La acción se produce al Visualizar
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar cuando se cumple la acción.
→ CAMPO_MENSAJENOOK	CADENA(200)	Mensaje a mostrar cuando la acción no se cumple.
→ CAMPO_TEXTO	CADENA(50)	Texto libre a pasar a la acción asociada a la transición

Métodos para aplicación de los filtros

→ obtenerAccionTransicion

2.4.2.1.3.5TrAmbitoLey

trewa.bd.trapi.tpo.TrAmbitoLey

Clase que representa los diferentes ámbitos de la ley para las normativas.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFAMBITOLEY	TpoPK	Identificador del ámbito de ley.
→ ABREVIATURA	String	Abreviatura del ámbito de ley.
→ DESCRIPCION	String	Descripción del ámbito de ley.
→ OBSOLETO	String	Indica si el ámbito de ley está obsoleto. → "S" – Sí → "N" – No
→ CODWANDA	String	Identificador del componente en w@ndA.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFAMBITOLEY	NUMÉRICO	Identificador del ámbito de ley.
→ CAMPO_ABREVIATURA	CADENA(10)	Abreviatura del ámbito de ley.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción del ámbito de ley.
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el ámbito de ley está obsoleto. → "S" – Sí → "N" – No
→ CAMPO_CODWANDA	CADENA(50)	Identificador del componente en w@ndA.

Métodos para aplicación de los filtros

→ obtenerAmbitoLey

2.4.2.1.3.6TrAviso

trewa.bd.trapi.tpo.TrAviso

Clase que representa la información de un aviso.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFAVISO	TpoPK	Identificador del aviso.
→ NOMBRE	String	Nombre con el que se denomina al aviso.
→ DESCRIPCION	String	Texto que describe el aviso.
→ STMA	TrSistema	Sistema.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFAVISO	NUMÉRICO	Identificador del aviso.
→ CAMPO_NOMBRE	CADENA(50)	Nombre con el que se denomina al aviso.
→ CAMPO_DESCRIPCION	CADENA(250)	Texto que describe el aviso.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema

Métodos para aplicación de los filtros

→ obtenerAviso

2.4.2.1.3.7TrAvisoBloquePermitido

trewa.bd.trapi.tpo.TrAvisoBloquePermitido

Clase que representa la información de un aviso asociado a una tarea en fase.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AVISO	TrAviso	Aviso.
→ BLOQUEPER	TrBloquePermitido	Bloque permitido.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ MENSAJEOK	String	Mensaje a mostrar.
→ VALIDA	String	Indica si el aviso asociado a la tarea es válido. → "S" – Sí → "N" – No
→ COMPROBAR	String	Indica si se realiza el aviso. → "I" – El aviso se produce al Incorporar → "V" – El aviso se produce al Visualizar

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFAVISO	NUMÉRICO	Identificador del aviso
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.

→ CAMPO_REFBLOQUEP ER	NUMÉRICO	Identificador del bloque permitido o tarea en fase
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase
→ CAMPO_REFBLOQUEIN I	NUMÉRICO	Identificador del bloque inicial o tarea llamante
→ CAMPO_REFBLOQUEFI N	NUMÉRICO	Identificador del bloque fin o tarea.
→ CAMPO_VALIDA	CADENA(1)	Indica si el aviso asociado a la tarea es válido. → “ S ” – Sí → “ N ” – No
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza el aviso. → “ I ” – El aviso se produce al Incorporar → “ V ” – El aviso se produce al Visualizar
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar.

Métodos para aplicación de los filtros

→ obtenerAvisoBloquePermitido

2.4.2.1.3.8TrAvisoDocumentoPermitido

trewa.bd.trapi.tpo.TrAvisoDocumentoPermitido

Clase que representa la información de un aviso asociado a un documento permitido.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AVISO	TrAviso	Aviso.
→ TIPODOC	TrTipoDocumento	Tipo de documento.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ FASE	TrFase	Fase.
→ MENSAJEOK	String	Mensaje a mostrar para el aviso.
→ VALIDA	String	Indica si el aviso asociado a la tarea es válido. → “ S ” – Sí → “ N ” – No
→ COMPROBAR	String	Indica si se realiza el aviso. → “ G ” – El aviso se produce al Generar el documento → “ I ” – El aviso se produce al Incorporar el documento → “ T ” – El aviso se produce al Generar e Incorporar el documento → “ V ” - El aviso se produce al Visualizar el documento

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFAVISO	NUMÉRICO	Identificador del aviso.
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase.
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar para el aviso.
→ CAMPO_VALIDA	CADENA(1)	Indica si el aviso asociado a la tarea es válido. → “ S ” – Sí → “ N ” – No
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza el aviso. → “ G ” – El aviso se produce al Generar el documento → “ I ” – El aviso se produce al Incorporar el documento → “ T ” – El aviso se produce al Generar e Incorporar el documento → “ V ” - El aviso se produce al Visualizar el documento

Métodos para aplicación de los filtros

→ obtenerAvisoDocumentoPermitido

2.4.2.1.3.9TrAvisoTransicion

trewa.bd.trapi.tpo.TrAvisoTransicion

Clase que representa la información de un aviso asociado a una transición.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AVISO	TrAviso	Aviso.
→ TRANSICION	TrTransicion	Transición.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ MENSAJEOK	String	Mensaje a mostrar para el aviso.
→ VALIDA	String	Indica si el aviso asociado a la transición es válido. → “ S ” – Sí → “ N ” – No
→ COMPROBAR	String	Indica si se realiza el aviso. → “ T ” – El aviso se produce al Tramitar → “ D ” – El aviso se produce al Deshacer → “ A ” – El aviso se produce al Tramitar y Deshacer

→ "V" – El aviso se produce al Visualizar

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFAVISO	NUMÉRICO	Identificador del aviso
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar para el aviso.
→ CAMPO_VALIDA	CADENA(1)	Indica si el aviso asociado a la transición es válido. → "S" – Sí → "N" – No
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza el aviso. → "T" – El aviso se produce al Tramitar → "D" – El aviso se produce al Deshacer → "A" – El aviso se produce al Tramitar y Deshacer → "V" – El aviso se produce al Visualizar

Métodos para aplicación de los filtros

→ obtenerAvisoTransicion

2.4.2.1.3.10TrBloque

trewa.bd.trapi.tpo.TrBloque

Clase que representa la información referente a un bloque.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFBLOQUE	TpoPK	Identificador del bloque.
→ NOMBRE	String	Nombre del bloque.
→ DESCRIPCION	String	Descripción del bloque.
→ TIPO	String	Tipo de bloque → "W" – Web → "R" – Informe → "F" – Pantalla → "O" – Otro

- | | | |
|-------------------|-----------|--|
| → INFORMAR | String | Indica si se informa al bus de esta tarea.
→ "S" – Sí
→ "N" – No |
| → STMA | TrSistema | Sistema. |

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFBLOQUE	NUMÉRICO	Identificador del bloque
→ CAMPO_NOMBRE	CADENA(50)	Nombre del bloque
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del bloque
→ CAMPO_TIPO	CADENA(1)	Tipo de bloque → "W" - Web → "R" – Informe → "F" – Pantalla → "O" – Otro
→ CAMPO_INFORMAR	CADENA(1)	Indica si se informa al bus de esta tarea. → "S" – Sí → "N" – No
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
Métodos para aplicación de los filtros		→ obtenerBloque

2.4.2.1.3.11 TrBloquePermitido

trewa.bd.trapi.tpo.TrBloquePermitido

Clase que representa la información referente a las tareas de tipo "MANIPULAR_DATOS" definidas en una fase de un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFBLOQUEPER	TpoPK	Identificador del bloque permitido o tarea en fase.
→ ETIQUETA	String	Etiqueta de la tarea en fase
→ DESCRIPCION	String	Descripción de la tarea en fase
→ VALIDO	String	Indica si es una tarea válida → "S" – Sí

→ ORDEN	Integer	→ "N" – No Indica el orden de la tarea en la fase
→ FASE	TrFase	Fase
→ BLOQUEINI	TrBloque	Bloque inicial o tarea llamante
→ BLOQUEFIN	TrBloque	Bloque final o tarea
→ OBLIGATORIO	String	Indica si es una tarea obligatoria → "S" – Sí → "N" – No
→ INFORMAR	String	Indica si se informa al bus de la tarea → "S" – Sí → "N" – No
→ ETIQLARGA	String	Etiqueta larga para la tarea

2.4.2.1.3.12TrBloquePermitidoDefProc

trewa.bd.trapi.tpo.TrBloquePermitidoDefProc

Clase que representa la información referente a las tareas de tipo "MANIPULAR_DATOS" asociadas a una definición de procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ BLOQUEPER	TrBloquePermitido	Bloque permitido.
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento.
→ UNIDAD	String	Unidad de medida para el plazo máximo de realización. → "D" – Días → "M" – Meses

- | | | |
|-----------------------|---------|--|
| → NUMUNIDADES | Integer | → “A” – Años
Indica el nº de unidades del plazo máximo de realización: nº de días, nº de meses, nº de años. |
| → DESCFECHALIM | String | Descripción del significado de la fecha límite de realización de la tarea. |

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→ CAMPO_UNIDAD	CADENA(1)	Unidad de medida para el plazo máximo de realización. → “D” – Días → “M” – Meses → “A” – Años
→ CAMPO_NUMUNIDADES	NUMÉRICO	Indica el nº de unidades del plazo máximo de realización: nº de días, nº de meses, nº de años.
→ CAMPO_DESCFECHALIM	CADENA(100)	Descripción del significado de la fecha límite de realización de la tarea.

Métodos para aplicación de los filtros

→ obtenerBloquePermitidoDefProc

2.4.2.1.3.13 TrBloquePermitidoPerfil

trewa.bd.trapi.tpo.TrBloquePermitidoPerfil

Clase que representa la información referente a un bloque asociado a un perfil de usuario.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ PERFILUSU	TrPerfilUsuario	Perfil de usuario.
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento.
→ REFFASE	TpoPK	Identificador de la fase.
→ REFBLOQUEINI	TpoPK	Identificador del bloque inicial.
→ REFBLOQUEFIN	TpoPK	Identificador del bloque final.

→ **REFBLOQUEPER** TpoPK Identificador del bloque permitido.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPERFILUS U	NUMÉRICO	Identificador del perfil de usuario.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase.
→ CAMPO_REFBLOQUEIN I	NUMÉRICO	Identificador del bloque inicial o tarea llamante.
→ CAMPO_REFBLOQUEFI N	NUMÉRICO	Identificador del bloque final o tarea.
→ CAMPO_REFBLOQUEP ER	NUMÉRICO	Identificador del bloque permitido o tarea en fase.

Métodos para aplicación de los filtros

→ obtenerBloquePermitidoPerfil

2.4.2.1.3.14TrCaducidad

trewa.bd.trapi.tpo.TrCaducidad

Clase que representa la información referente a caducidades.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFCADUCIDAD	TpoPK	Identificador de la caducidad.
→ ABREVIATURA	String	Abreviatura de la caducidad.
→ DESCRIPCION	String	Descripción de la caducidad.
→ UNIDAD	String	Unidad de medida para el plazo máximo de realización. → "D" – Días → "M" – Meses → "A" – Años
→ NUMUNIDADES	Integer	Indica el nº de unidades del plazo máximo de realización: nº de días, nº de meses, nº de años.
→ TIPO	String	Indica el tipo de caducidad. → "P" – Plazo de ejecución → "O" – Otro
→ VIGENTE	String	Indica si la caducidad está vigente. → "S" – Sí → "N" – No
→ TRANSICION	TrTransicion	Transición.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCADUCIDA D	NUMÉRICO	Identificador de la caducidad
→ CAMPO_ABREVIATUR A	CADENA(10)	Abreviatura de la caducidad
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción de la caducidad
→ CAMPO_UNIDAD	CADENA(1)	Unidad de medida para el plazo máximo de realización. → "D" – Días → "M" – Meses → "A" – Años
→ CAMPO_UNIDADES	NUMÉRICO	Indica el nº de unidades del plazo máximo de realización: nº de días, nº de meses, nº de años.
→ CAMPO_TIPO	CADENA(1)	Indica el tipo de caducidad. → "P" – Plazo de ejecución → "O" – Otro
→ CAMPO_VIGENTE	CADENA(1)	Indica si la caducidad está vigente. → "S" – Sí → "N" – No
→ CAMPO_REFTRANSICI ON	NUMÉRICO	Identificador de la transición
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
Métodos para aplicación de los filtros		→ obtenerCaducidad

2.4.2.1.3.15TrCertificado

trewa.bd.trapi.tpo.TrCertificado

Clase que representa los certificados configurados para el componente @firma.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFCERT	TpoPK	Identificador global del certificado.
→ NOMB_CERTIFICADO	String	Nombre del certificado.
→ TDESCRIPCION	String	Descripción del certificado.
→ ALIAS	String	Alias del certificado.
→ PASSWORD	String	Contraseña del certificado o del almacén del certificado.
→ PASSWORDALIAS	String	Contraseña del certificado dentro del almacén
→ RUTA	String	Ruta donde se encuentra el certificado/almacén
→ VIGENTE	String	Indica si el certificado está vigente o no. → "S" – Sí → "N" – No
→ LTIPO	String	Indica el tipo del certificado: almacén (JKS) ó certificado privado (P12, PFX,...)

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CERT	NUMÉRICO	Identificador del certificado
→ CAMPO_TIPO	CADENA(6)	Indica el tipo de certificado.
→ CAMPO_NOMBRE	CADENA(128)	Identificador de la definición del procedimiento.
→ CAMPO_ALIAS	CADENA(32)	Indica si la condición o acción se debe cumplir obligatoriamente o no. → "S" – Sí → "N" – No
→ CAMPO_VIGENTE	CADENA(1)	Indica si el certificado está vigente o no. → "S" – Sí → "N" – No

Métodos para aplicación de los filtros

→ obtenerCertificados

2.4.2.1.3.16TrComponente

trewa.bd.trapi.tpo.TrComponente

Clase que representa los componentes que pueden intervenir en el sistema w@ndA.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFCOMPONENTE	TpoPK	Identificador global del componente.
→ NOMBRE	String	Nombre del componente.
→ DESCRIPCION	String	Descripción del componente.
→ DIRECCIONIP	String	Dirección IP del componente.

→ TIPOCOMPONENTE	TrTipoComponente	Tipo de componente.
→ USUARIO	String	Usuario para acceder al componente.
→ PASSWORD	String	Password del usuario para acceder al componente.
→ CODWANDA	Long	Identificador del componente en w@ndA.
→ REFORGANISMO	TpoPK	Identificador del organismo donde se encuentra el componente

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCOMPONENTE	NUMÉRICO	Identificador global del componente.
→ CAMPO_NOMBRE	CADENA(50)	Nombre del componente.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del componente.
→ CAMPO_DIRECCIONIP	CADENA(25)	Dirección IP del componente.
→ CAMPO_REFTIPOCOMPONENTE	NUMÉRICO	Identificador del tipo de componente
→ CAMPO_USUARIO	CADENA(30)	Usuario para acceder al componente.
→ CAMPO_PASSWORD	CADENA(30)	Password del usuario para acceder al componente.
→ CAMPO_CODWANDA	NUMÉRICO	Identificador del componente en w@ndA.
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo donde se encuentra el componente

Métodos para aplicación de los filtros

→ obtenerComponente

2.4.2.1.3.17TrCondicionAccion

trewa.bd.trapi.tpo.TrCondicionAccion

Clase que representa la información referente a una condición o acción

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFCONDACC	TpoPK	Identificador de la condición o acción
→ NOMBRE	String	Nombre con el que se denomina la condición o acción
→ DESCRIPCION	String	Descripción de la condición o acción
→ PARAMREFEXP	String	Indica si se le pasa el id del expediente a la condición o acción. → "S" – Sí → "N" – No
→ PARAMREFTRAN	String	Indica si se le pasa el id de la transición a la condición o acción. → "S" – Sí → "N" – No

→ PARAMREFDOCPER	String	Indica si se le pasa el id del paso en la evolución actual a la condición o acción. → “ S ” – Sí → “ N ” – No
→ PARAMREFEXPFAS	String	Indica si se le pasa el id del paso en la evolución actual a la condición o acción. → “ S ” – Sí → “ N ” – No
→ PARAMREFDEFPROC	String	Indica si se le pasa el id de la definición del procedimiento a la condición o acción. → “ S ” – Sí → “ N ” – No
→ PARAMFECHA	String	Indica si se le pasa la fecha de tramitación/generación/incorporación indicada a la condición o acción. → “ S ” – Sí → “ N ” – No
→ PARAMUSUARIO	String	Indica si se le pasa el usuario a la condición o acción. → “ S ” – Sí → “ N ” – No
→ PARAMREFFASE	String	Indica si se le pasa el identificador de la fase a la condición o acción. → “ S ” – Sí → “ N ” – No
→ PARAMREFTIPODOC	String	Indica si se le pasa el identificador del tipo de documento a la condición o acción. → “ S ” – Sí → “ N ” – No
→ IMPLEMENTACION	String	Indica el tipo de implementación asociada a la condición o acción. → “ F ” – Función en el servicios → “ J ” – Java
→ PAQUETE	String	Indica el paquete que contiene la función a ejecutar si se trata de una función en el servidor, o el nombre del paquete y clase al que pertenece la función java a ejecutar.
→ NOMBFUNCIÓN	String	Indica el nombre de la función a ejecutar.
→ COMPLEJA	String	Indica si la condición es compleja. → “ S ” – Sí → “ N ” – No
→ EXPRESION	String	Contiene la expresión a evaluar en el caso de expresiones complejas.
→ STMA	TrSistema	Sistema de la condición o acción.
→ TIPO	String	Indica el tipo → “ C ” – Condición → “ A ” – Acción → “ W ” – Aviso
→ TEXTO	String	Texto libre a pasar a la condición / acción

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCONDACC	NUMÉRICO	Identificador de la condición o acción.
→ CAMPO_NOMBRE	CADENA(50)	Nombre con el que se denomina la condición o acción.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la condición o acción.

→ CAMPO_PARAMREFEX P	CADENA(1)	Indica si se le pasa el id del expediente a la condición o acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFTRAN	CADENA(1)	Indica si se le pasa el id de la transición a la condición o acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFDOCUMENTO	CADENA(1)	Indica si se le pasa el id del documento permitido a la condición o acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFEX PXFAS	CADENA(1)	Indica si se le pasa el id del paso en la evolución actual a la condición o acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFDEFINICION	CADENA(1)	Indica si se le pasa el id de la definición del procedimiento a la condición o acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMFECHA	CADENA(1)	Indica si se le pasa la fecha de tramitación/generación/incorporación indicada a la condición o acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMUSUARIO	CADENA(1)	Indica si se le pasa el usuario a la condición o acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFFASE	CADENA(1)	Indica si se le pasa el identificador de la fase a la condición o acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFTIPO	CADENA(1)	Indica si se le pasa el identificador del tipo de documento a la condición o acción. → “ S ” – Sí → “ N ” – No
→ CAMPO_IMPLEMENTACION	CADENA(1)	Indica el tipo de implementación asociada a la condición o acción. → “ F ” – Función en el servidor → “ J ” – Java
→ CAMPO_PAQUETE	CADENA(100)	Indica el paquete que contiene la función a ejecutar si se trata de una función en el servidor, o el nombre del paquete y clase al que pertenece la función java a ejecutar.
→ CAMPO_NOMB_FUNCION	CADENA(50)	Indica el nombre de la función a ejecutar.
→ CAMPO_COMPLEJA	CADENA(1)	Indica si la condición es compleja. → “ S ” – Sí → “ N ” – No
→ CAMPO_EXPRESION	CADENA(4000)	Contiene la expresión a evaluar en el caso de expresiones complejas.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema.
→ CAMPO_TIPO	CADENA(1)	Indica el tipo → “ C ” – Condición → “ A ” – Acción → “ W ” – Aviso
→ CAMPO_TEXTO	CADENA(50)	Texto libre a pasar a la condición / acción

Métodos para aplicación de los filtros

→ obtenerCondAccAvi

2.4.2.1.3.18TrCondAccionBloquePermitido

trewa.bd.trapi.tpo. TrCondAccionBloquePermitido

Clase que representa la información de una condición o acción asociada a una tarea en fase.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CONDACC	TrCondicionAccion	Condición o acción.
→ BLOQUEPER	TrBloquePermitido	Bloque permitido.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ MENSAJEOK	String	Mensaje a mostrar cuando se cumple la condición o acción.
→ MENSAJENOOK	String	Mensaje a mostrar cuando la condición o acción no se cumple.
→ OBLIGATORIA	String	Indica si la condición o acción se debe cumplir obligatoriamente o no. → “ S ” – Sí → “ N ” – No
→ VALIDA	String	Indica si la condición o acción asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ MOSTRARMSJ	String	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ COMPROBAR	String	Indica si se realiza la condición o acción. → “ I ” – La condición se produce al Incorporar → “ V ” – La condición se produce al Visualizar
→ TEXTO	String	Texto libre a pasar a la condición / acción asociada al bloque permitido

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCONDADD	NUMÉRICO	Identificador de la condición o acción
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→ CAMPO_REFBLOQUEPER	NUMÉRICO	Identificador del bloque permitido o tarea en fase
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase.
→ CAMPO_REFBLOQUEIN	NUMÉRICO	Identificador del bloque inicial o tarea llamante

I		
→ CAMPO_REFBLOQUEFI	NUMÉRICO	Identificador del bloque final o tarea
N		
→ CAMPO_OBLIGATORIA	CADENA(1)	Indica si la condición o acción se debe cumplir obligatoriamente o no. → “ S ” – Sí → “ N ” – No
→ CAMPO_VALIDA	CADENA(1)	Indica si la condición o acción asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza la condición o acción. → “ I ” – La condición o acción se produce al Incorporar → “ V ” – La condición o acción se produce al Visualizar
→ CAMPO_MOSTRARMSJ	CADENA(1)	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar cuando se cumple la condición o acción.
→ CAMPO_MENSAJENOO	CADENA(200)	Mensaje a mostrar cuando no se cumple la condición o acción.
K		
→ CAMPO_TEXTO	CADENA(50)	Texto libre a pasar a la condición / acción asociada al bloque permitido

Métodos para aplicación de los filtros

→ obtenerCondAccAviBloquePermitido

2.4.2.1.3.19TrCondAccionDocumentoPermitido

trewa.bd.trapi.tpo. TrCondAccionDocumentoPermitido

Clase que representa la información de una condición o acción asociada a un documento permitido.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CONDACC	TrCondicionAccion	Condición o acción.
→ TIPODOC	TrTipoDocumento	Tipo de documento.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ FASE	TrFase	Fase.
→ MENSAJEOK	String	Mensaje a mostrar cuando se cumple la condición o acción.
→ MENSAJENOOK	String	Mensaje a mostrar cuando la condición o acción no se cumple.
→ OBLIGATORIA	String	Indica si la condición o acción se debe cumplir obligatoriamente o no. → “ S ” – Sí → “ N ” – No
→ VALIDA	String	Indica si la condición o acción asociada a la tarea es válida.

→ MOSTRARMSJ	String	→ “ S ” – Sí → “ N ” – No Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ COMPROBAR	String	Indica si se realiza la condición o acción. → “ G ” – La condición o acción se produce al Generar el documento → “ I ” – La condición o acción se produce al Incorporar el documento → “ T ” – La condición o acción se produce al Generar e Incorporar el documento → “ V ” – La condición o acción se produce al Visualizar el documento
→ TEXTO	String	Texto libre a pasar a la condición / acción asociada al documento permitido

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCONDACC	NUMÉRICO	Identificador de la condición o acción.
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase.
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar cuando se cumple la condición o acción.
→ CAMPO_MENSAJENOO	CADENA(200)	Mensaje a mostrar cuando la condición o acción no se cumple.
K		
→ CAMPO_OBLIGATORIA	CADENA(1)	Indica si la condición o acción se debe cumplir obligatoriamente o no. → “ S ” – Sí → “ N ” – No
→ CAMPO_VALIDA	CADENA(1)	Indica si la condición o acción asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ CAMPO_MOSTRARMSJ	CADENA(1)	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza la condición o acción. → “ G ” – La condición o acción se produce al Generar el documento → “ I ” – La condición o acción se produce al Incorporar el documento → “ T ” – La condición o acción se produce al Generar e Incorporar el documento → “ V ” – La condición o acción se produce al Visualizar el documento
→ CAMPO_TEXTO	CADENA(50)	Texto libre a pasar a la condición / acción asociada al documento permitido

Métodos para aplicación de los filtros

→ obtenerCondAccAviDocumentoPermitido

2.4.2.1.3.20TrCondAccionTransicion

trewa.bd.trapi.tpo. TrCondAccionTransicion

Clase que representa la información de una condición o acción de una transición.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CONDACC	TrCondicionAccion	Condición o acción.
→ TRANSICION	TrTransicion	Transición.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ MENSAJEOK	String	Mensaje a mostrar cuando se cumple la condición o acción.
→ MENSAJENOOK	String	Mensaje a mostrar cuando la condición o acción no se cumple.
→ OBLIGATORIA	String	Indica si la condición o acción se debe cumplir obligatoriamente o no. → "S" – Sí → "N" – No
→ VALIDA	String	Indica si la condición o acción asociada a la tarea es válida. → "S" – Sí → "N" – No
→ MOSTRARMSJ	String	Mensaje a mostrar. → "O" – Mensaje OK → "N" – Mensaje No OK → "A" – Ambos → "I" – Ninguno
→ COMPROBAR	String	Indica si se realiza la condición o acción. → "T" – La condición o acción se produce al Tramitar → "D" – La condición o acción se produce al Deshacer → "A" – La condición o acción se produce al Tramitar y Deshacer → "V" – La condición o acción se produce al Visualizar
→ TEXTO	String	Texto libre a pasar a la condición / acción asociada a la transición

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCONDACC	NUMÉRICO	Identificador de la condición o acción.
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.

- **CAMPO_OBLIGATORIA** CADENA(1) Indica si la condición o acción se debe cumplir obligatoriamente o no.
→ "S" – Sí
→ "N" – No
- **CAMPO_VALIDA** CADENA(1) Indica si la condición o acción asociada a la tarea es válida.
→ "S" – Sí
→ "N" – No
- **CAMPO_MOSTRARMSJ** CADENA(1) Mensaje a mostrar.
→ "O" – Mensaje OK
→ "N" – Mensaje No OK
→ "A" – Ambos
→ "I" – Ninguno
- **CAMPO_COMPROBAR** CADENA(1) Indica si se realiza la condición o acción.
→ "T" – La condición o acción se produce al Tramitar
→ "D" – La condición o acción se produce al Deshacer
→ "A" – La condición o acción se produce al Tramitar y Deshacer
→ "V" – La condición o acción se produce al Visualizar
- **CAMPO_MENSAJEOK** CADENA(200) Mensaje a mostrar cuando se cumple la condición.
- **CAMPO_MENSAJENOOK** CADENA(200) Mensaje a mostrar cuando no se cumple la condición o acción.
- **CAMPO_TEXTO** CADENA(50) Texto libre a pasar a la condición / acción asociada a la transición

Métodos para aplicación de los filtros

→ obtenerCondAccAviTransicion

2.4.2.1.3.21 TrCondicion

trewa.bd.trapi.tpo.TrCondicion

Clase que representa la información referente a una condición.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFCONDICION	TpoPK	Identificador de la condición.
→ NOMBRE	String	Nombre con el que se denomina la condición.
→ DESCRIPCION	String	Descripción de la condición.
→ PARAMREFEXP	String	Indica si se le pasa el id del expediente a la condición. → “ S ” – Sí → “ N ” – No
→ PARAMREFTRAN	String	Indica si se le pasa el id de la transición a la condición. → “ S ” – Sí → “ N ” – No
→ PARAMREFDOCPER	String	Indica si se le pasa el id del paso en la evolución actual a la condición. → “ S ” – Sí → “ N ” – No
→ PARAMREFEXPXNAS	String	Indica si se le pasa el id del paso en la evolución actual a la condición. → “ S ” – Sí → “ N ” – No
→ PARAMREFDEFPROC	String	Indica si se le pasa el id de la definición del procedimiento a la condición. → “ S ” – Sí → “ N ” – No
→ PARAMFECHA	String	Indica si se le pasa la fecha de tramitación/generación/incorporación indicada a la condición. → “ S ” – Sí → “ N ” – No
→ PARAMUSUARIO	String	Indica si se le pasa el usuario a la condición. → “ S ” – Sí → “ N ” – No
→ PARAMREFFASE	String	Indica si se le pasa el identificador de la fase a la condición. → “ S ” – Sí → “ N ” – No

→ PARAMREFTIPODOC	String	Indica si se le pasa el identificador del tipo de documento a la condición. → “ S ” – Sí → “ N ” – No
→ IMPLEMENTACION	String	Indica el tipo de implementación asociada a la condición. → “ F ” – Función en el servidores → “ J ” – Java
→ PAQUETE	String	Indica el paquete que contiene la función a ejecutar si se trata de una función en el servidor, o el nombre del paquete y clase al que pertenece la función java a ejecutar.
→ NOMBFUNCION	String	Indica el nombre de la función a ejecutar.
→ COMPLEJA	String	Indica si la condición es compleja. → “ S ” – Sí → “ N ” – No
→ EXPRESION	String	Contiene la expresión a evaluar en el caso de expresiones complejas.
→ STMA	TrSistema	Sistema de la condición.
→ TEXTO	String	Texto libre a pasar a la condición

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCONDICION	NUMÉRICO	Identificador de la condición.
→ CAMPO_NOMBRE	CADENA(50)	Nombre con el que se denomina la condición.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la condición.
→ CAMPO_PARAMREFEX	CADENA(1)	Indica si se le pasa el id del expediente a la condición. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFTR	CADENA(1)	Indica si se le pasa el id de la transición a la condición. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFDO	CADENA(1)	Indica si se le pasa el id del documento permitido a la condición. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFEX	CADENA(1)	Indica si se le pasa el id del paso en la evolución actual a la condición. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFDE	CADENA(1)	Indica si se le pasa el id de la definición del procedimiento a la condición. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMFECHA	CADENA(1)	Indica si se le pasa la fecha de tramitación/generación/incorporación indicada a la condición. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMUSUAR	CADENA(1)	Indica si se le pasa el usuario a la condición. → “ S ” – Sí → “ N ” – No

→ CAMPO_PARAMREFFASE	CADENA(1)	Indica si se le pasa el identificador de la fase a la condición. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFTIPODOC	CADENA(1)	Indica si se le pasa el identificador del tipo de documento a la condición. → “ S ” – Sí → “ N ” – No
→ CAMPO_IMPLEMENTACION	CADENA(1)	Indica el tipo de implementación asociada a la condición. → “ F ” – Función en el servidores → “ J ” – Java
→ CAMPO_PAQUETE	CADENA(100)	Indica el paquete que contiene la función a ejecutar si se trata de una función en el servidor, o el nombre del paquete y clase al que pertenece la función java a ejecutar.
→ CAMPO_NOMBFUNCION	CADENA(50)	Indica el nombre de la función a ejecutar.
→ CAMPO_COMPLEJA	CADENA(1)	Indica si la condición es compleja. → “ S ” – Sí → “ N ” – No
→ CAMPO_EXPRESION	CADENA(4000)	Contiene la expresión a evaluar en el caso de expresiones complejas.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema.
→ CAMPO_TEXTO	CADENA(50)	Texto libre a pasar a la condición

Métodos para aplicación de los filtros

→ obtenerCondicion

2.4.2.1.3.22TrCondicionBloquePermitido

trewa.bd.trapi.tpo.TrCondicionBloquePermitido

Clase que representa la información de una condición asociada a una tarea en fase.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CONDICION	TrCondicion	Condición.
→ BLOQUEPER	TrBloquePermitido	Bloque permitido.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ MENSAJEOK	String	Mensaje a mostrar cuando se cumple la condición.
→ MENSAJENOOK	String	Mensaje a mostrar cuando la condición no se cumple.
→ OBLIGATORIA	String	Indica si la condición se debe cumplir obligatoriamente o no. → “ S ” – Sí → “ N ” – No
→ VALIDA	String	Indica si la condición asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ MOSTRARMSJ	String	Mensaje a mostrar.

→ COMPROBAR	String	→ “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno Indica si se realiza la condición. → “ I ” – La condición se produce al Incorporar → “ V ” – La condición se produce al Visualizar
→ TEXTO	String	Texto libre a pasar a la condición asociada al bloque permitido

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCONDICION	NUMÉRICO	Identificador de la condición
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→ CAMPO_REFBLOQUEPER	NUMÉRICO	Identificador del bloque permitido o tarea en fase
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase.
→ CAMPO_REFBLOQUEIN	NUMÉRICO	Identificador del bloque inicial o tarea llamante
→ CAMPO_REFBLOQUEFIN	NUMÉRICO	Identificador del bloque final o tarea
→ CAMPO_OBLIGATORIA	CADENA(1)	Indica si la condición se debe cumplir obligatoriamente o no. → “ S ” – Sí → “ N ” – No
→ CAMPO_VALIDA	CADENA(1)	Indica si la condición asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza la condición. → “ I ” – La condición se produce al Incorporar → “ V ” – La condición se produce al Visualizar
→ CAMPO_MOSTRARMSJ	CADENA(1)	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar cuando se cumple la condición.
→ CAMPO_MENSAJENOO	CADENA(200)	Mensaje a mostrar cuando no se cumple la condición.
→ CAMPO_TEXTO	CADENA(50)	Texto libre a pasar a la condición asociada al bloque permitido.

Métodos para aplicación de los filtros

→ obtenerCondicionBloquePermitido

2.4.2.1.3.23TrCondicionDocumentoPermitido

trewa.bd.trapi.tpo.TrCondicionDocumentoPermitido

Clase que representa la información de una condición asociada a un documento permitido.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CONDICION	TrCondicion	Condición.
→ TIPODOC	TrTipoDocumento	Tipo de documento.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ FASE	TrFase	Fase.
→ MENSAJEOK	String	Mensaje a mostrar cuando se cumple la condición.
→ MENSAJENOOK	String	Mensaje a mostrar cuando la condición no se cumple.
→ OBLIGATORIA	String	Indica si la condición se debe cumplir obligatoriamente o no. → "S" – Sí → "N" – No

→ VALIDA	String	Indica si la condición asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ MOSTRARMSJ	String	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ COMPROBAR	String	Indica si se realiza la condición. → “ G ” – La condición se produce al Generar el documento → “ I ” – La condición se produce al Incorporar el documento → “ T ” – La condición se produce al Generar e Incorporar el documento → “ V ” – La condición se produce al Visualizar el documento
→ TEXTO	String	Texto libre a pasar a la condición asociada al documento permitido

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCONDICION N	NUMÉRICO	Identificador de la condición.
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase.
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar cuando se cumple la condición.
→ CAMPO_MENSAJENOO K	CADENA(200)	Mensaje a mostrar cuando la condición no se cumple.
→ CAMPO_OBLIGATORIA	CADENA(1)	Indica si la condición se debe cumplir obligatoriamente o no. → “ S ” – Sí → “ N ” – No
→ CAMPO_VALIDA	CADENA(1)	Indica si la condición asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ CAMPO_MOSTRARMSJ	CADENA(1)	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza la condición. → “ G ” – La condición se produce al Generar el documento → “ I ” – La condición se produce al Incorporar el documento → “ T ” – La condición se produce al Generar e Incorporar el documento → “ V ” – La condición se produce al Visualizar el documento
→ CAMPO_TEXTO	CADENA(50)	Texto libre a pasar a la condición asociada al documento permitido

Métodos para aplicación de los filtros

→ obtenerCondicionDocumentoPermitido

2.4.2.1.3.24TrCondicionTransicion

trewa.bd.trapi.tpo.TrCondicionTransicion

Clase que representa la información de una condición de una transición.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CONDICION	TrCondicion	Condición.
→ TRANSICION	TrTransicion	Transición.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ MENSAJEOK	String	Mensaje a mostrar cuando se cumple la condición.
→ MENSAJENOOK	String	Mensaje a mostrar cuando la condición no se cumple.

→ OBLIGATORIA	String	Indica si la condición se debe cumplir obligatoriamente o no. → “ S ” – Sí → “ N ” – No
→ VALIDA	String	Indica si la condición asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ MOSTRARMSJ	String	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ COMPROBAR	String	Indica si se realiza la condición. → “ T ” – La condición se produce al Tramitar → “ D ” – La condición se produce al Deshacer → “ A ” – La condición se produce al Tramitar y Deshacer → “ V ” – La condición se produce al Visualizar
→ TEXTO	String	Texto libre a pasar a la condición asociada a la transición

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFCONDICION	NUMÉRICO	Identificador de la condición.
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_OBLIGATORIA	CADENA(1)	Indica si la condición se debe cumplir obligatoriamente o no. → “ S ” – Sí → “ N ” – No
→ CAMPO_VALIDA	CADENA(1)	Indica si la condición asociada a la tarea es válida. → “ S ” – Sí → “ N ” – No
→ CAMPO_MOSTRARMSJ	CADENA(1)	Mensaje a mostrar. → “ O ” – Mensaje OK → “ N ” – Mensaje No OK → “ A ” – Ambos → “ I ” – Ninguno
→ CAMPO_COMPROBAR	CADENA(1)	Indica si se realiza la condición. → “ T ” – La condición se produce al Tramitar → “ D ” – La condición se produce al Deshacer → “ A ” – La condición se produce al Tramitar y Deshacer → “ V ” – La condición se produce al Visualizar
→ CAMPO_MENSAJEOK	CADENA(200)	Mensaje a mostrar cuando se cumple la condición.
→ CAMPO_MENSAJENOOK	CADENA(200)	Mensaje a mostrar cuando no se cumple la condición.
→ CAMPO_TEXTO	CADENA(50)	Texto libre a pasar a la condición asociada a la transición

Métodos para aplicación de los filtros

→ obtenerCondicionTransicion

2.4.2.1.3.25TrConsPlantMult

trewa.bd.trapi.tpo.TrConsPlantMult

Clase que representa la información de una consulta de una plantilla múltiple.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFCONSPANTMULT	TpoPK	Identificador de consulta.
→ NOMBRE	String	Nombre de la consulta.
→ DESCRIPCION	String	Descripción de la consulta.

→ PARAMREFEXP	String	Indica si se le pasa id de expediente como parámetro a la consulta. → “ S ” – Sí → “ N ” – No
→ PARAMREFDOC PER	String	Indica si se le pasa id de documento permitido como parámetro a la consulta. → “ S ” – Sí → “ N ” – No
→ PARAMREFEXPX FAS	String	Indica si se le pasa id de expediente en fase como parámetro a la consulta. → “ S ” – Sí → “ N ” – No
→ PARAMREFDEF PROC	String	Indica si se le pasa id de procedimiento como parámetro a la consulta. → “ S ” – Sí → “ N ” – No
→ PARAMFECHA	String	Indica si se le pasa la fecha de tramitación como parámetro a la consulta. → “ S ” – Sí → “ N ” – No
→ STMA	String	Sistema
→ PARAMUSUARIO	String	Indica si se le pasa id de usuario como parámetro a la consulta. → “ S ” – Sí → “ N ” – No
→ PARAMREFTIPO DOC	String	Indica si se le pasa id de tipodocumento como parámetro a la consulta. → “ S ” – Sí → “ N ” – No
→ PAQUETE	String	Indica nombre de paquete donde se encuentra el paquete y clase de la función
→ NOMB FUNCION	String	Nombre de la función a ejecutar para obtener la consulta

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REF_CAMPO	NUMÉRICO	Identificador de la consulta sobre la plantilla.
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la consulta sobre plantilla.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción de la consulta sobre la plantilla.
→ CAMPO_PARAMREFEXP	CADENA(1)	Indica si se le pasa id de expediente como parámetro a la consulta. → “ S ” – Sí → “ N ” – No
→ CAMPO_PARAMREFDOC PER	CADENA(1)	Indica si se le pasa id de documento permitido como parámetro a la consulta.

- **CAMPO_PARAMREFEX** CADENA(1)
PXFAS
→ “S” – Sí
→ “N” – No
Indica si se le pasa id de expediente en fase como parámetro a la consulta.
- **CAMPO_PARAMREFDE** CADENA(1)
FPROC
→ “S” – Sí
→ “N” – No
Indica si se le pasa id de procedimiento como parámetro a la consulta.
- **CAMPO_PARAMFECHA** CADENA(1)
→ “S” – Sí
→ “N” – No
Indica si se le pasa la fecha de tramitación como parámetro a la consulta.
- **CAMPO_PARAMREFST** CADENA(1)
MA
→ “S” – Sí
→ “N” – No
Indica si se le pasa id de usuario como parámetro a la consulta.
- **CAMPO_PARAMUSUARIO** CADENA(1)
IO
→ “S” – Sí
→ “N” – No
Indica si se le pasa id de tipodocumento como parámetro a la consulta.
- **CAMPO_PARAMREFTIP** CADENA(1)
ODOC
→ “S” – Sí
→ “N” – No
Indica nombre de paquete donde se encuentra el paquete y clase de la función
- **CAMPO_PAQUETE** CADENA(50)
Nombre de la función a ejecutar para obtener la consulta
- **CAMPO_NOMBFUNCION** CADENA(50)
N

Métodos para aplicación de los filtros

→ obtenerConsultaPlantillaMultiple

2.4.2.1.3.26TrConstante

trewa.bd.trapi.tpo.TrConstante

Clase que recoge la información de las constantes del motor de tramitación.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODCONSTANTE	String	Código de la constante.
→ DESCRIPCION	String	Descripción de la constante.
→ VALOR	String	Valor de la constante.
→ STMA	TrSistema	Sistema.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODIGOCTE	CADENA(20)	Código de la constante.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la constante.
→ CAMPO_VALOR	CADENA(500)	Valor de la constante.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema.

Métodos para aplicación de los filtros → obtenerConstante

2.4.2.1.3.27TrConstanteGn

trewa.bd.trapi.tpo.TrConstanteGn

Clase que recoge la información de las constantes generales.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODCONSTANTE	String	Código de la constante general.
→ DESCRIPCION	String	Descripción de la constante general.
→ VALOR	String	Valor de la constante general.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODCONSTANTE	CADENA(10)	Código de la constante general.
→ CAMPO_DESCRIPCION	CADENA(64)	Descripción de la constante general.
→ CAMPO_VALOR	CADENA(80)	Valor de la constante general.

Métodos para aplicación de los filtros → obtenerConstanteGeneral

2.4.2.1.3.28TrDatoComponente

trewa.bd.trapi.tpo.TrDatoComponente

Clase que recoge otros datos necesarios para el componente (atributo-valor).

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDATOCOMP	TpoPK	Identificador del dato del componente.
→ COMPONENTE	TrComponente	Componente.
→ ATRIBUTO	String	Nombre del atributo del componente.
→ VALOR	String	Valor del atributo del componente.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDATOCOMP	NUMÉRICO	Identificador del dato del componente.
→ CAMPO_REFCOMP	NUMÉRICO	Identificador del componente.
→ CAMPO_ATRIBUTO	CADENA(32)	Nombre del atributo del componente.
→ CAMPO_VALOR	CADENA(255)	Valor del atributo del componente.

Métodos para aplicación de los filtros

→ obtenerDatoComponente

2.4.2.1.3.29TrDefProcedimiento

trewa.bd.trapi.tpo.TrDefProcedimiento

Clase que representa la información de una definición de un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoría
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento.
→ ABREVIATURA	String	Abreviatura de la definición del procedimiento.
→ DESCRIPCION	String	Descripción de la definición del procedimiento.
→ STMA	TrSistema	Sistema de la definición del procedimiento.
→ INFORMAR	String	Indica si se tendrá que informar al bus de este procedimiento y de los expedientes que se tramiten. → "S" – Sí → "N" – No
→ BLOQUEADO	String	Indica si el procedimiento está bloqueado. → "S" – Sí

→ USUARIOBLQ	String	→ "N" – No Usuario que tiene bloqueado el procedimiento
→ VIGENTE	String	Indica si el procedimiento está vigente. → "S" – Sí → "N" – No
→ CATEGORIA	String	Indica si es Familia, subfamilia o procedimiento. → "F" – Familia → "S" – Subfamilia → "P" – Procedimiento
→ DESCRIPCIONAMP	String	Descripción ampliada del procedimiento.
→ REFDEFPROCPADRE	TpoPK	Identificador de la familia/subfamilia.
→ ORGANISMO	TrOrganismo	Organismo a la que pertenece.
→ ORGCOMPETENTE	TrOrganismo	Organismo competente.
→ ORGRESUELVE	TrOrganismo	Organismo que resuelve.
→ ORGTRAMITA	TrOrganismo	Organismo que tramita.
→ CODWANDA	String	Valor en w@ndA del procedimiento
→ COMENTARIOS	String	Comentarios al procedimiento/familia/subfamilia
→ FINALIDAD	String	Finalidad de la definición de procedimiento, para soporte SCSP.
→ CODIGORPA	String	Código RPA del procedimiento para certificados SCSP
→ SERIEDOCARCH	String	Serie documental del procedimiento para @rchiVA
→ ADAPTADOENI	String	Indica si el procedimiento se encuentra adaptado para ENI. → "S" – Sí → "N" – No
→ CLASIFICACION	String	Clasificación del procedimiento o código SIA de procedimiento
→ SERVICIONOTI	String	Código del servicio de notifica asociado al procedimiento
→ QAA	String	Nivel con el que se accede a la información de un expediente
→ NIVELINTCC	String	Nivel de integración con Carpeta Ciudadana
→ REFCERT	TpoPK	Identificador del certificado de @firma asociado

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_ABREVIATUR A	CADENA(10)	Abreviatura de la definición del procedimiento.
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción de la definición del procedimiento.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema de la definición del procedimiento.
→ CAMPO_INFORMAR	CADENA(1)	Indica si se tendrá que informar al bus de este procedimiento y de los expedientes que se tramiten. → "S" – Sí → "N" – No
→ CAMPO_USUARIOBLQ	CADENA(10)	Indica el usuario que tiene bloqueado el procedimiento
→ CAMPO_VIGENTE	CADENA(1)	Indica si el procedimiento está vigente. → "S" – Sí → "N" – No
→ CAMPO_CATEGORIA	CADENA(1)	Indica si es Familia, subfamilia o procedimiento. → "F" – Familia

→ CAMPO_DESCRIPCION AMP	CADENA(250)	→ "S" – Subfamilia → "P" – Procedimiento Descripción ampliada del procedimiento.
→ CAMPO_REFDEFPROC PADRE	NUMÉRICO	Identificador de la familia/subfamilia.
→ CAMPO_REFORGANISMO	NUMÉRICO	Organismo al que pertenece.
→ CAMPO_REFCERTIFICADO	NUMÉRICO	Certificado asociado al procedimiento
→ CAMPO_REFORGCAMP	NUMÉRICO	Organismo competente.
→ CAMPO_REFORGRES	NUMÉRICO	Organismo que resuelve.
→ CAMPO_REFORGTRAM	NUMÉRICO	Organismo que tramita.
→ CAMPO_CODWANDA	CADENA(15)	Valor en w@ndA del procedimiento
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICA	DATE	Fecha de modificación del registro
→ CAMPO_COMENTARIOS	CADENA	Comentarios asociados al procedimiento
→ CAMPO_FINALIDAD	CADENA	Finalidad del procedimiento
→ CAMPO_CODIGORPA	CADENA	Código RPA del procedimiento.

Métodos para aplicación de los filtros

→ obtenerDefProcedimiento

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.2.1.3.30TrDefProcedimientoGr

trewa.bd.trapi.tpo.TrDefProcedimientoGr

Clase que representa la información de una definición de un procedimiento gráfico.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDEFPROCGR	TpoPK	Identificador de la definición gráfica del procedimiento.
→ DESCDIAGRAMA	String	Descripción de la definición gráfica del procedimiento.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROCGR	NUMÉRICO	Identificador de la definición gráfica del procedimiento.
→ CAMPO_DESCDIAGRAMA	CADENA(250)	Descripción de la definición gráfica del procedimiento.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.

Métodos para aplicación de los filtros

→ obtenerDefProcedimientoGr

2.4.2.1.3.31TrDiaFestivo

trewa.bd.trapi.tpo.TrDiaFestivo

Clase que representa la información de un día festivo

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDIAFESTIVO	TpoPK	Identificador del día festivo
→ CODPROVINCIA	String	Código de la provincia
→ PROVINCIA	String	Descripción de la provincia
→ CODMUNICIPIO	String	Código del municipio
→ MUNICIPIO	String	Descripción del municipio
→ FECHA	Timestamp	Fecha del festivo
→ DESCRIPCION	String	Descripción del festivo

→ **PERIODICO** String Indica si es periódico o no.
Sus valores pueden ser:
→ “**S**” – Sí
→ “**N**” – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDIAFESTIVO	NUMÉRICO	Identificador del día festivo
→ CAMPO_CODPROVINCIA	CADENA	Código de la provincia
→ CAMPO_PROVINCIA	CADENA	Descripción de la provincia
→ CAMPO_CODMUNICIPIO	CADENA	Código del municipio
→ CAMPO_MUNICIPIO	CADENA	Descripción del municipio
→ CAMPO_FECHA	FECHA	Fecha del festivo
→ CAMPO_DESCRIPCION	CADENA	Descripción del festivo
→ CAMPO_PERIODICO	CADENA	Indica si es periódico o no. Sus valores pueden ser: → “ S ” – Sí → “ N ” – No

Métodos para aplicación de los filtros → obtenerDiaFestivo

2.4.2.1.3.32 TrDocumentoDelegado

trewa.bd.trapi.tpo.TrDocumentoDelegado

Clase que representa los documentos para los que los delegados de firma tienen competencia para firmar.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDOCDEL	TpoPK	Identificador del documento delegado.
→ FECHAINIVIG	Timestamp	Fecha de inicio de vigencia para la delegación del documento.
→ FECHAFINVIG	Timestamp	Fecha fin de vigencia para la delegación del documento.
→ FIRMDEF	TrFirmanteDefinido	Firmante definido.
→ TIPODOC	TrTipoDocumento	Tipo de documento.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDOCDEL	NUMÉRICO	Identificador del documento delegado.
→ CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia para la delegación del documento.
→ CAMPO_FECHAFINVIG	DATE	Fecha fin de vigencia para la delegación del documento.
→ CAMPO_REFFIRMDEF	NUMÉRICO	Identificador del firmante definido.
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento.
<i>Métodos para aplicación de los filtros</i>		→ obtenerDocumentoDelegado

2.4.2.1.3.33 TrDocumentoPermitido

trewa.bd.trapi.tpo.TrDocumentoPermitido

Clase que representa los documentos permitidos en una determinada fase.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDOCPER	TpoPK	Identificador del documento permitido.
→ OBLIGATORIO	String	Indica si el documento permitido es obligatorio. → "S" – Sí → "N" – No
→ VALIDO	String	Indica si el documento permitido es válido
→ ORDEN	Integer	Indica el orden del documento permitido en la fase.
→ TIPODOC	TrTipoDocumento	Tipo de documento

→ FASE	TrFase	Fase.
→ INFORMAR	String	Indica si se informa al bus del documento → “ S ” – Sí → “ N ” – No
→ ETIQUETA	String	Texto con el que se etiqueta al documento permitido en la fase
→ DESCRIPCION	String	Descripción de lo que representa el documento permitido en la fase
→ ETIQLARGA	String	Etiqueta larga para el documento permitido

2.4.2.1.3.34TrDocumentoPermitidoDefProc

trewa.bd.trapi.tpo.TrDocumentoPermitidoDefProc

Clase que representa los documentos permitidos en una determinada fase que se asocian a un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ DOCPER	TrDocumentoPermitido	Documento permitido.
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento.
→ UNIDAD	String	Unidad de medida para el plazo máximo de realización. → “ D ” – Días → “ M ” – Meses → “ A ” – Años
→ NUMUNIDADES	Integer	Indica el nº de unidades del plazo máximo de realización: nº de días, nº de meses, nº de años.
→ DESCFECHALIM	String	Descripción del significado de la fecha límite de realización de la tarea.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_UNIDAD	CADENA(1)	Unidad de medida para el plazo máximo de realización. → “ D ” – Días → “ M ” – Meses → “ A ” – Años
→ CAMPO_NUMUNIDADE S	NUMÉRICO	Indica el nº de unidades del plazo máximo de realización: nº de días, nº de meses, nº de años.
→ CAMPO_DESCFECHALI M	CADENA(100)	Descripción del significado de la fecha límite de realización de la tarea.

Métodos para aplicación de los filtros

→ obtenerDocumentoPermitidoDefProc

2.4.2.1.3.35TrDocumentoPermitidoPerfil

trewa.bd.trapi.tpo.TrDocumentoPermitidoPerfil

Clase que representa los documentos permitidos por procedimiento y perfil de usuario.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ PERMISO	String	Permiso para el perfil. Sus valores pueden ser: → “ G ” – Generar → “ I ” – Incorporar → “ F ” – Firmar → “ E ” – Editar → “ T ” – Todo
→ PERFILUSU	TrPerfilUsuario	Perfil de usuario.
→ REFTIPODOC	TpoPK	Identificador del tipo de documento.
→ REFFASE	TpoPK	Identificador de la fase.
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_PERMISO	CADENA(1)	Permiso para el perfil. Sus valores pueden ser: → “ G ” – Generar → “ I ” – Incorporar → “ F ” – Firmar → “ E ” – Editar → “ T ” – Todo
→ CAMPO_REFPERFILUSU	NUMÉRICO	Identificador del perfil de usuario
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento
→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.

Métodos para aplicación de los filtros

→ obtenerDocumentoPermitidoPerfil

2.4.2.1.3.36TrEmpleado

trewa.bd.trapi.tpo.TrEmpleado

Clase que representa los usuarios que ocupan los puestos de trabajo en los organismos definidos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoría
→ USUARIO	TrUsuario	Usuario.
→ ORGANISMO	TrOrganismo	Organismo.
→ PTOTRABAJO	TrPuestoTrabajo	Puesto de trabajo.
→ TIPO	String	Indica el tipo de empleado. → "F" – Funcionario → "L" – Laboral
→ FECHANOMBRA MIENTO	Timestamp	Fecha de ocupación del puesto por el usuario.
→ FECHACESE	Timestamp	Fecha de baja del usuario en el puesto.
→ TRATAMIENTO	String	Texto de tratamiento para los documentos (D., D ^a , Ilmo., etc.)
→ TXREFERENCIA	String	Texto de referencia para el usuario.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_USUARIO	CADENA(10)	Código del usuario.
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo.
→ CAMPO_CODPTOTRAB	CADENA(10)	Código del puesto de trabajo.
→ CAMPO_TIPO	CADENA(1)	Indica el tipo de empleado. → "F" – Funcionario → "L" – Laboral
→ CAMPO_FECHANOMBRA MIENTO	DATE	Fecha de ocupación del puesto por el usuario.
→ CAMPO_FECHACESE	DATE	Fecha de baja del usuario en el puesto

→ CAMPO_TRATAMIENTO	CADENA(64)	Texto de tratamiento para los documentos (D., D ^a , Ilmo., etc.)
→ CAMPO_TXTREFERENCIA	CADENA(10)	Texto de referencia para el usuario.
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICA	DATE	Fecha de modificación del registro

Métodos para aplicación de los filtros

→ obtenerEmpleado

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.2.1.3.37TrError

trewa.bd.trapi.tpo.TrError

Clase que representa los errores definidos para el tramitador.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFERROR	TpoPK	Identificador del error.
→ MENSAJE	String	Mensaje del error.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFERROR	NUMÉRICO	Identificador del error.

→ **CAMPO_MENSAJE** CADENA(250) Mensaje del error

Métodos para aplicación de los filtros

→ obtenerError

2.4.2.1.3.38TrExtremoTransicionGr

trewa.bd.trapi.tpo.TrExtremoTransicionGr

Clase que representa la información gráfica de los distintos extremos de transición que pueden darse.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFEXTREMOGR	TpoPK	Identificador del extremo gráfico.
→ XIZQ	Integer	Componente X de la coordenada superior izquierda del gráfico.
→ YARR	Integer	Componente Y de la coordenada superior izquierda del gráfico.
→ ANCHO	Integer	Ancho del gráfico.
→ ALTO	Integer	Alto del gráfico.
→ DEFPROCGR	TrDefProcedimientoGr	Definición del procedimiento gráfico.
→ FASE	TrFase	Fase.
→ TIPO	String	Indica el tipo de extremo de la transición. → " FA " – Fase normal → " I " – Fase inicial → " D " – División → " U " – Unión → " F " – Fase final → " ES " – Evento salida, abandona la fase actual. → " EN " – Evento que no abandona la fase actual.
→ METAFASEGR	TrMetafaseGr	Metafase gráfica.
→ COLORFONDO	Long	Color de fondo del extremo de la transición.
→ COLORTEXTO	Long	Color del texto del extremo de la transición.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFEXTREMO	NUMÉRICO	Identificador del extremo gráfico.



GR

- **CAMPO_XIZQ** NUMÉRICO Componente X de la coordenada superior izquierda del gráfico.
- **CAMPO_YARR** NUMÉRICO Componente Y de la coordenada superior izquierda del gráfico.
- **CAMPO_ANCHO** NUMÉRICO Ancho del gráfico.
- **CAMPO_ALTO** NUMÉRICO Alto del gráfico.
- **CAMPO_REFDEFPROC** NUMÉRICO Identificador de la definición del procedimiento gráfico.

GR

- **CAMPO_REFFASE** NUMÉRICO Identificador de la fase.
- **CAMPO_TIPO** CADENA(2) Indica el tipo de extremo de la transición.
 - **"FA"** – Fase normal
 - **"I"** – Fase inicial
 - **"D"** – División
 - **"U"** – Unión
 - **"F"** – Fase final
 - **"ES"** – Evento salida, abandona la fase actual.
 - **"EN"** – Evento que no abandona la fase actual.
- **CAMPO_REFMETAFAS** NUMÉRICO Identificador de la metafase gráfica.

EGR

- **CAMPO_COLORFONDO** NUMÉRICO Color de fondo del extremo de la transición.
- **CAMPO_COLORTEXTO** NUMÉRICO Color del texto del extremo de la transición.

Métodos para aplicación de los filtros

→ obtenerExtremoTransicionGr

2.4.2.1.3.39TrFase

trewa.bd.trapi.tpo.TrFase

Clase que representa la información de una fase.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoría	Datos de auditoría
→ REFFASE	TpoPK	Identificador de la fase.
→ NOMBRE	String	Nombre de la fase.
→ DESCRIPCION	String	Descripción de la fase.
→ ORDEN	Integer	Orden no riguroso de la fase dentro de la metafase.
→ DEFPROC	TrDefProcedimientoq	Definición del procedimiento.
→ METAFASE	TrMetafase	Metafase a la que pertenece la fase.
→ STMA	TrSistema	Sistema al que está asociada la fase.
→ INFORMAR	String	Indica si se informa al bus de la entrada en la fase. → "S" – Sí → "N" – No
→ TEXTOAUXILIAR	String	Texto auxiliar para la fase

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
---------------------	--------------------	--

→ CAMPO_REFFASE	NUMÉRICO	Identificador de la fase.
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la fase.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la fase.
→ CAMPO_ORDEN	NUMÉRICO	Orden no riguroso de la fase dentro de la metafase.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_REFMETAFAS	NUMÉRICO	Identificador de la metafase a la que pertenece la fase.
E		
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema al que está asociada la fase.
→ CAMPO_INFORMAR	CADENA(1)	Indica si se informa al bus de la entrada en la fase. → "S" – Sí → "N" – No
→ CAMPO_TEXTOAUXILI	CADENA(500)	Texto auxiliar para la fase.
AR		
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREAD	DATE	Fecha de creación del registro
O		
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFI	DATE	Fecha de modificación del registro
CA		

Métodos para aplicación de los filtros

→ obtenerFase

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.2.1.3.40TrFirma

trewa.bd.trapi.tpo.TrFirma

Clase que representa la firma de los usuarios firmantes.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFFIRMA	TpoPK	Identificador de la firma.
→ FIRMA	byte[]	Gráfico de la firma.
→ USUARIO	String	Usuario que firma.
→ FORMATO	String	Tipo de formato de la imagen (tipo mime).
→ NOMBREFICHERO	String	Nombre del fichero de la imagen.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFFIRMA	NUMÉRICO	Identificador de la firma.
→ CAMPO_USUARIO	CADENA(10)	Usuario que firma.
→ CAMPO_FORMATO	CADENA(128)	Tipo de formato de la imagen (tipo mime).
→ CAMPO_NOMBREFICHERO	CADENA(64)	Nombre del fichero de la imagen.

Métodos para aplicación de los filtros

→ obtenerFirma

2.4.2.1.3.41 TrFirmanteDefinido

trewa.bd.trapi.tpo.TrFirmanteDefinido

Clase que representa los firmantes principales, sustitutos y delegados definidos en el sistema.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFFIRMDEF	TpoPK	Identificador del firmante definido.
→ TIPO	String	Tipo de firmante. → " P " – Principal → " D " – Delegado → " S " – Sustituto
→ FECHAINIVIG	Timestamp	Fecha de inicio de vigencia del firmante definido.
→ FECHAFINVIG	Timestamp	Fecha de fin de vigencia del firmante definido.

- **TEXTODISPOSICION** TrTextoDisposicionFirma Texto de la disposición.
- **REFDELEGSUST** TpoPK Identificador del delegado o sustituto.
- **PTOTRABAJO** TrPuestoTrabajo Puesto de trabajo.
- **ORGANISMO** TrOrganismo Organismo.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFFIRMDEF	NUMÉRICO	Identificador del firmante definido.
→ CAMPO_TIPO	CADENA(1)	Tipo de firmante. → "P" – Principal → "D" – Delegado → "S" – Sustituto
→ CAMPO_FECHAINIVIGE	DATE	Fecha de inicio de vigencia del firmante definido.
→ CAMPO_FECHAFINIG	DATE	Fecha de fin de vigencia del firmante definido.
→ CAMPO_TEXTODISPOSICION	CADENA(10)	Texto de la disposición.
→ CAMPO_REFDELEGSUST	NUMÉRICO	Identificador del del delegado o sustituto.
→ CAMPO_CODPTOTRAB	CADENA(10)	Código del puesto de trabajo.
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo.

Métodos para aplicación de los filtros

→ obtenerFirmanteDefinido

2.4.2.1.3.42TrFirmaTipoDocumento

trewa.bd.trapi.tpo.TrFirmaTipoDocumento

Clase que representa las firmas que puede tener un tipo de documento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ TIPODOC	TrTipoDocumento	Tipo de documento.
→ FIRMDEF	TrFirmanteDefinido	Firmante definido.
→ ORDEN	Integer	Orden de la firma.

- **EDITABLE** String Indica si el pie de firma que corresponde con la firma es editable o no.
→ "**S**" – Sí
→ "**N**" – No
- **ETIQUETA** String Etiqueta del pie de firma.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento.
→ CAMPO_REFFIRMDEF	NUMÉRICO	Identificador del firmante definido.
→ CAMPO_ORDEN	NUMÉRICO	Orden de la firma.
→ CAMPO_EDITABLE	CADENA(1)	Indica si el pie de firma que corresponde con la firma es editable o no. → " S " – Sí → " N " – No
→ CAMPO_ETIQUETA	CADENA(20)	Etiqueta del pie de firma.

Métodos para aplicación de los filtros

→ obtenerFirmaTipoDocumento

2.4.2.1.3.43TrIndicacionFicha

trewa.bd.trapi.tpo.TrIndicacionFicha

Clase que representa datos variables sobre la ficha descriptiva del procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFINDFICHA	TpoPK	Identificador de la indicación de la ficha del procedimiento.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.

- **TIPOINDICA** TrTipoIndicacion Tipo de indicación de la ficha.
- **DESCRIPCION** String Descripción de la indicación de la ficha del procedimiento.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFINDFICHA	NUMÉRICO	Identificador de la indicación de la ficha del procedimiento.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_REFTIPOINDIC A	NUMÉRICO	Identificador del tipo de indicación de la ficha.
→ CAMPO_DESCRIPCION	CADENA(300)	Descripción de la indicación de la ficha del procedimiento.

Métodos para aplicación de los filtros

→ obtenerIndicacionFicha

2.4.2.1.3.44TrLimiteCaducidad

trewa.bd.trapi.tpo.TrLimiteCaducidad

Clase que representa el límite para una caducidad.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ TRANSICION	TrTransicion	Transición.
→ CADUCIDAD	TrCaducidad	Caducidad.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición.
→ CAMPO_REFCADUCIDAD	NUMÉRICO	Identificador de la caducidad.

Métodos para aplicación de los filtros

→ obtenerLimiteCaducidad

2.4.2.1.3.45TrMetafase

trewa.bd.trapi.tpo.TrMetafase

Clase que representa una agrupación de fases según el significado en la tramitación.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoría
→ REFMETAFASE	TpoPK	Identificador de la metafase.
→ NOMBRE	String	Nombre de la metafase.
→ DESCRIPCION	String	Descripción de la metafase.
→ ORDEN	Integer	Orden no riguroso de la metafase dentro del procedimiento en el que aparecen.
→ INFORMAR	String	Indica si se informa al bus de la entrada en la metafase. → "S" – Sí → "N" – No
→ STMA	TrSistema	Sistema.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFMETAFASE	NUMÉRICO	Identificador de la metafase.
→ CAMPO_NOMBRE	CADENA(50)	Nombre de la metafase.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la metafase.
→ CAMPO_ORDEN	NUMÉRICO	Orden no riguroso de la metafase dentro del procedimiento en el que aparecen.
→ CAMPO_INFORMAR	CADENA(1)	Indica si se informa al bus de la entrada en la metafase. → "S" – Sí → "N" – No
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREACION	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICACION	DATE	Fecha de modificación del registro

Métodos para aplicación de los filtros

→ obtenerMetafase

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.2.1.3.46TrMetafaseGr

trewa.bd.trapi.tpo.TrMetafaseGr

Clase que representa la información gráfica de una metafase.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFMETAFASEGR	TpoPK	Identificador de la metafase gráfica.
→ XIZQ	Integer	Componente X de la coordenada superior izquierda del gráfico.
→ YARR	Integer	Componente Y de la coordenada superior izquierda del gráfico.
→ ANCHO	Integer	Ancho del gráfico.
→ ALTO	Integer	Alto del gráfico.
→ DEFPROCGR	TrDefProcedimientoGr	Definición del procedimiento gráfico.
→ METAFASE	TrMetafase	Metafase.
→ COLORFONDO	Long	Color de fondo del gráfico de metafase.
→ COLORTEXTO	Long	Color del texto del gráfico de metafase.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFMETAFAS EGR	NUMÉRICO	Identificador de la metafase gráfica.
→ CAMPO_XIZQ	NUMÉRICO	Componente X de la coordenada superior izquierda del gráfico
→ CAMPO_YARR	NUMÉRICO	Componente Y de la coordenada superior izquierda del gráfico
→ CAMPO_ANCHO	NUMÉRICO	Ancho del gráfico
→ CAMPO_ALTO	NUMÉRICO	Alto del gráfico
→ CAMPO_REFDEFPROC GR	NUMÉRICO	Identificador de la definición del procedimiento gráfico.
→ CAMPO_REFMETAFAS E	NUMÉRICO	Identificador de la metafase.
→ CAMPO_COLORFONDO	NUMÉRICO	Color de fondo del gráfico de metafase.
→ CAMPO_COLORTEXTO	NUMÉRICO	Color del texto del gráfico de metafase.

Métodos para aplicación de los filtros

→ obtenerMetafaseGr

2.4.2.1.3.47TrMunicipio

trewa.bd.trapi.tpo.TrMunicipio

Clase que representa la información de un municipio de una provincia.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODMUNICIPIO	String	Código del municipio.
→ PROVINCIA	TrProvincia	Provincia a la que pertenece el municipio.
→ NOMBRE	String	Nombre del municipio.
→ DESCRIPCION	String	Descripción del municipio.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODMUNICIPIO	CADENA(3)	Código del municipio.
→ CAMPO_CODPROVINCIA	CADENA(2)	Código de la provincia a la que pertenece el municipio.
→ CAMPO_NOMBRE	CADENA(32)	Nombre del municipio.
→ CAMPO_DESCRIPCION	CADENA(64)	Descripción del municipio.

Métodos para aplicación de los filtros

→ obtenerMunicipio

2.4.2.1.3.48TrNodoTransicionGr

trewa.bd.trapi.tpo.TrNodoTransicionGr

Clase que representa a información gráfica de los nodos de una transición.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFNODOTRANGR	TpoPK	Identificador del nodo de la transición gráfica.
→ X	Integer	Componente X de la coordenada del nodo.
→ Y	Integer	Componente Y de la coordenada del nodo.
→ ORDEN	Integer	Orden del nodo en un gráfico de transición.
→ TRANSGR	TrTransicionGr	Transición gráfica.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFNODOTRANGR	NUMÉRICO	Identificador del nodo de la transición gráfica.
→ CAMPO_X	NUMÉRICO	Componente X de la coordenada del nodo
→ CAMPO_Y	NUMÉRICO	Componente Y de la coordenada del nodo
→ CAMPO_ORDEN	NUMÉRICO	Orden del nodo en un gráfico de transición
→ CAMPO_REFTRANGR	NUMÉRICO	Identificador de la transición gráfica.

Métodos para aplicación de los filtros

→ obtenerNodoTransicionGr

2.4.2.1.3.49TrNormativa

trewa.bd.trapi.tpo.TrNormativa

Clase que representa las distintas normativas que rigen los procedimientos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFNORMATIVA	TpoPK	Identificador de la normativa.
→ TIPONORMA	TrTipoNormativa	Tipo de normativa.
→ TIPOPUBLI	TrTipoPublicacion	Tipo de publicación.
→ DESCNORMATIVA	String	Descripción de la normativa.
→ FEHAVIGOR	Timestamp	Fecha de entrada en vigor de la normativa.
→ TITULO	String	Título de la normativa.
→ NUMERO	Long	Número de la normativa.
→ FECHAPUBLI	Timestamp	Fecha de publicación de la normativa.
→ NUMPUBLI	Long	Número de publicación de la normativa.
→ ANYO	Integer	Año de la normativa.
→ AMBITOLEY	TrAmbitoLey	Ámbito de la ley de la normativa.
→ CODWANDA	String	Valor en w@ndA de la normativa

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFNORMATIVA	NUMÉRICO	Identificador de la normativa.
→ CAMPO_REFTIPONORMA	NUMÉRICO	Identificador del tipo de normativa.
→ CAMPO_REFTIPOPUBLI	NUMÉRICO	Identificador del tipo de publicación.
→ CAMPO_DESCNORMATIVA	CADENA(250)	Descripción de la normativa.
→ CAMPO_FEHAVIGOR	DATE	Fecha de entrada en vigor de la normativa.
→ CAMPO_TITULO	CADENA(50)	Título de la normativa
→ CAMPO_NUMERO	NUMÉRICO	Número de la normativa.
→ CAMPO_FECHAPUBLI	DATE	Fecha de publicación de la normativa.
→ CAMPO_NUMPUBLI	NUMÉRICO	Número de publicación de la normativa.
→ CAMPO_CODWANDA	CADENA(15)	Valor en w@ndA de la normativa
→ CAMPO_ANYO	NUMÉRICO	Año de la normativa.
→ CAMPO_REFAMBITOLEY	NUMÉRICO	Identificador del ámbito de la ley de la normativa.

Métodos para aplicación de los filtros

→ obtenerNormativa

2.4.2.1.3.50TrNormativaDefProcedimiento

trewa.bd.trapi.tpo.TrNormativaDefProcedimiento

Clase que representa las distintas normativas por las que se rige un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento.
→ NORMATIVA	TrNormativa	Normativa.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_REFNORMATIVA	NUMÉRICO	Identificador de la normativa.
→ CAMPO_REFTIPONORMA	NUMÉRICO	Identificador del tipo de normativa.
→ CAMPO_REFTIPOPUBLI	NUMÉRICO	Identificador del tipo de publicación.
→ CAMPO_DESCNORMATIVA	CADENA(250)	Descripción de la normativa.
→ CAMPO_FECHAVIGOR	DATE	Fecha de entrada en vigor de la normativa.
→ CAMPO_TITULO	CADENA(50)	Título de la normativa
→ CAMPO_NUMERO	NUMÉRICO	Número de la normativa.
→ CAMPO_FECHAPUBLI	DATE	Fecha de publicación de la normativa.
→ CAMPO_NUMPUBLI	NUMÉRICO	Número de publicación de la normativa.
→ CAMPO_CODWANDA	CADENA(15)	Valor en w@ndA de la normativa
→ CAMPO_ANYO	NUMÉRICO	Año de la normativa.
→ CAMPO_REFAMBITOLEY	NUMÉRICO	Identificador del ámbito de la ley de la normativa.

Métodos para aplicación de los filtros

→ obtenerNormativaDefProcedimiento

2.4.2.1.3.51TrOrganismo

trewa.bd.trapi.tpo.TrOrganismo

Clase que representa la información de los distintos organismos o unidades orgánicas.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoría
→ REFORGANISMO	TpoPK	Identificador del organismo.
→ CODMUNICIPIO	String	Código del municipio.
→ CODPROVINCIA	String	Código de la provincia.
→ TIPOVIA	String	Tipo de vía.
→ CODORG	String	Código del organismo.
→ NOMBRE	String	Nombre del organismo.
→ DESCRIPCION	String	Descripción del organismo.
→ TIPO	String	Tipo de organismo. → " DC " – Delegación Provincial → " SC " – Servicios Centrales → " CE " – Centro Adscrito → " EE " – Empresa Externa
→ NOMBREVIA	String	Nombre de la vía pública del domicilio.
→ NUMERO	Integer	Número del domicilio.
→ LETRA	String	Letra del domicilio.
→ ESCALERA	String	Escalera del domicilio.
→ PISO	Integer	Piso del domicilio.
→ PUERTA	Stringq	Puerta del domicilio.
→ CODPOSTAL	Integer	Código postal del domicilio.
→ TELEFONO	String	Número(s) de teléfono de contacto.
→ TLFMOVIL	String	Número(s) de teléfono móvil.
→ FAX	String	Número(s) de Fax.

→ EMAIL	String	Correo electrónico.
→ CODCIWA	String	Código de identificación w@ndA.
→ IDARIES	String	Identificador de aries.
→ IDENTIFICADOR	String	Número de identificación (CIF).
→ DIGCONTROL	String	Dígito de control para el número de identificación.
→ FECHAINIVIG	Timestamp	Fecha de inicio de vigencia del organismo.
→ FECHAFINIVIG	Timestamp	Fecha de fin de vigencia del organismo
→ REFORGPADRE	TpoPK	Identificador del organismo superior al que pertenece.
→ TIPOORG	TrTipoOrganismo	Tipo de organismo.
→ TIPOORGZ	TrTipoOrganizacion	Tipo de organización.
→ ORDENPRELACION	String	Orden de relación del organismo
→ IMAGEN	byte[]	Imagen del organismo
→ DIR3	String	Código DIR3 del organismo
→ REFCERT	TpoPK	Identificador del certificado asociado al organismo

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo.
→ CAMPO_CODMUNICIPIO	CADENA(3)	Código del municipio.
→ CAMPO_CODPROVINCIA	CADENA(2)	Código de la provincia.
→ CAMPO_TIPOVIA	CADENA(10)	Código del tipo de vía.
→ CAMPO_CODORG	CADENA(16)	Código del organismo
→ CAMPO_NOMBRE	CADENA(32)	Nombre del organismo.
→ CAMPO_DESCRIPCION	CADENA(200)	Descripción del organismo.
→ CAMPO_TIPO	CADENA(2)	Tipo de organismo. → " DC " – Delegación Provincial → " SC " – Servicios Centrales → " CE " – Centro Adscrito → " EE " – Empresa Externa
→ CAMPO_NOMBREVIA	CADENA(64)	Nombre de la vía pública del domicilio.
→ CAMPO_NUMERO	NUMÉRICO	Número del domicilio.
→ CAMPO_LETRA	CADENA(2)	Letra del domicilio.
→ CAMPO_ESCALERA	CADENA(2)	Escalera del domicilio.
→ CAMPO_PISO	NUMÉRICO	Piso del domicilio.
→ CAMPO_PUERTA	CADENA(2)	Puerta del domicilio.
→ CAMPO_CODPOSTAL	NUMÉRICO	Código postal del domicilio.
→ CAMPO_TELEFONO	CADENA(25)	Número(s) de teléfono de contacto.
→ CAMPO_TLFMOVIL	CADENA(25)	Número(s) de teléfono móvil.
→ CAMPO_FAX	CADENA(25)	Número(s) de Fax.
→ CAMPO_EMAIL	CADENA(64)	Correo electrónico.
→ CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA.
→ CAMPO_IDARIES	CADENA(5)	Identificador de aries.
→ CAMPO_IDENTIFICADO	CADENA(15)	Número de identificación (CIF).
→ CAMPO_DIGCONTROL	CADENA(1)	Dígito de control para el número de identificación.

→ CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia del organismo.
→ CAMPO_FECHAFINIVIG	DATE	Fecha de fin de vigencia del organismo
→ CAMPO_REFORGPADRE	NUMÉRICO	Identificador del organismo superior al que pertenece.
→ CAMPO_REFTIPOORG	NUMÉRICO	Identificador del tipo de organismo.
→ CAMPO_REFTIPOORGZ	NUMÉRICO	Identificador del tipo de organización.
→ CAMPO_ORDENPRELACION	CADENA(15)	Orden de relación del organismo
→ CAMPO_REFCERT	NUMÉRICO	Identificador del certificado asociado.
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREADO	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICACION	DATE	Fecha de modificación del registro

Métodos para aplicación de los filtros

→ obtenerOrganismo

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.2.1.3.52TrPais

trewa.bd.trapi.tpo.TrPais

Clase que representa la información de un país.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODPAIS	String	Código del país.
→ NOMBRE	String	Nombre del país.
→ DESCRIPCION	String	Descripción del país.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODPAIS	CADENA(3)	Código del país.
→ CAMPO_NOMBRE	CADENA(32)	Nombre del país.
→ CAMPO_DESCRIPCION	CADENA(64)	Descripción del país.

Métodos para aplicación de los filtros

→ obtenerPais

2.4.2.1.3.53TrParametro

trewa.bd.trapi.tpo.TrParametro

Clase que representa los parámetros definidos en el sistema.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPARAM	TpoPK	Identificador del parámetro.
→ NOMBRE	String	Nombre del parámetro.
→ DESCRIPCION	String	Descripción del parámetro.
→ TIPO	String	Tipo del parámetro. → " N " – Numérico → " C " – Cadena → " F " – Fecha
→ TAMANIO	Integer	Tamaño del parámetro.
→ STMA	TrSistema	Sistema.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPARAM	NUMÉRICO	Identificador del parámetro.
→ CAMPO_NOMBRE	CADENA(20)	Nombre del parámetro.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción del parámetro.
→ CAMPO_TIPO	CADENA(1)	Tipo del parámetro. → " N " – Numérico → " C " – Cadena → " F " – Fecha
→ CAMPO_TAMANIO	NUMÉRICO	Tamaño del parámetro.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema

Métodos para aplicación de los filtros

→ obtenerParametro

2.4.2.1.3.54 TrParametroBloque

trewa.bd.trapi.tpo.TrParametroBloque

Clase que representa un parámetro que se define para un bloque.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ PARAMETRO	TrParametro	Parámetro.
→ BLOQUE	TrBloque	Bloque.
→ ORDEN	Integer	Orden del parámetro.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPARAM	NUMÉRICO	Identificador del parámetro.
→ CAMPO_REFBLOQUE	NUMÉRICO	Identificador del bloque.
→ CAMPO_ORDEN	NUMÉRICO	Orden del parámetro.

Métodos para aplicación de los filtros

→ obtenerParametroBloque

2.4.2.1.3.55 TrParametroVariable

trewa.bd.trapi.tpo.TrParametroVariable

Clase que representa un parámetro que se define para una variable.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ PARAMETRO	TrParametro	Parámetro.
→ VARIABLE	TrVariable	Variable.
→ ORDEN	Integer	Orden del parámetro.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPARAM	NUMÉRICO	Identificador del parámetro.
→ CAMPO_REFVARIABLE	NUMÉRICO	Identificador de la variable.
→ CAMPO_ORDEN	NUMÉRICO	Orden del parámetro.

Métodos para aplicación de los filtros

→ obtenerParametroVariable

2.4.2.1.3.56TrParrafoTipoDocumento

trewa.bd.trapi.tpo.TrParrafoTipoDocumento

Clase que representa un párrafo para un tipo de documento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPARRTIPODOC	TpoPK	Identificador del párrafo del tipo de documento.
→ PARRAFO	String	Texto del párrafo.
→ ORDEN	Integer	Orden del párrafo dentro del documento.
→ ALINEACION	String	Alineación del párrafo. → "I" – Izquierda → "C" – Centrado → "D" – Derecha → "J" – Justificado
→ ETIQUETA	String	Etiqueta del párrafo.
→ EDITABLE	String	Indica si el párrafo es editable. → "S" – Sí → "N" – No
→ ESTILO	String	Estilo del texto del párrafo. → "NORMAL" – Normal → "NEGRIT" – Negrita → "CURSIV" – Cursiva → "SUBRAY" – Subrayado → "NE-CU" – Negrita y Cursiva → "NE-SU" – Negrita y Subrayado → "CU-SU" – Cursiva y Subrayado → "NE-CU-SU" – Negrita, Cursiva y Subrayado
→ ESTILOETIQ	String	Estilo de la etiqueta del párrafo. → "NORMAL" – Normal → "NEGRIT" – Negrita → "CURSIV" – Cursiva → "SUBRAY" – Subrayado → "NE-CU" – Negrita y Cursiva → "NE-SU" – Negrita y Subrayado

→ TIPODOC	TrTipoDocumento	→ " CU-SU " – Cursiva y Subrayado
→ TIPOPARR	TrTipoParrafo	→ " NE-CU-SU " – Negrita, Cursiva y Subrayado
→ FUSIONAR	String	Tipo de documento.
		Tipo de párrafo.
		Indica si se fusionan o no variables al generar.
		→ " S " – Sí
		→ " N " – No
→ IMAGEN	byte[]	Imagen del párrafo.
→ FORMATO	String	Tipo de formato de la imagen (tipo mime).
→ NOMBREFICHERO	String	Nombre del fichero del documento físico.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPARRTIPO DOC	NUMÉRICO	Identificador del párrafo del tipo de documento.
→ CAMPO_PARRAFO	CADENA	Texto del párrafo
→ CAMPO_ORDEN	NUMÉRICO	Orden del párrafo dentro del documento.
→ CAMPO_ALINEACION	CADENA(1)	Alineación del párrafo. → " I " – Izquierda → " C " – Centrado → " D " – Derecha → " J " – Justificado
→ CAMPO_ETIQUETA	CADENA(15)	Etiqueta del párrafo.
→ CAMPO_EDITABLE	CADENA(1)	Indica si el párrafo es editable. → " S " – Sí → " N " – No
→ CAMPO_ESTILO	CADENA(8)	Estilo del texto del párrafo. → " NORMAL " – Normal → " NEGRIT " – Negrita → " CURSIV " – Cursiva → " SUBRAY " – Subrayado → " NE-CU " – Negrita y Cursiva → " NE-SU " – Negrita y Subrayado → " CU-SU " – Cursiva y Subrayado → " NE-CU-SU " – Negrita, Cursiva y Subrayado
→ CAMPO_ESTILOETIQ	CADENA(8)	Estilo de la etiqueta del párrafo. → " NORMAL " – Normal → " NEGRIT " – Negrita → " CURSIV " – Cursiva → " SUBRAY " – Subrayado → " NE-CU " – Negrita y Cursiva → " NE-SU " – Negrita y Subrayado → " CU-SU " – Cursiva y Subrayado → " NE-CU-SU " – Negrita, Cursiva y Subrayado
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador el tipo de documento.
→ CAMPO_REFTIPOPARR	NUMÉRICO	Identificador del tipo de párrafo.
→ CAMPO_FUSIONAR	CADENA(1)	Indica si se fusionan o no variables al generar.

→ CAMPO_FORMATO	CADENA(128)	→ "S" – Sí
→ CAMPO_NOMBREFICHERO	CADENA(64)	→ "N" – No
		Tipo de formato de la imagen (tipo mime)
		Nombre del fichero del documento físico

Métodos para aplicación de los filtros

→ obtenerParrafoTipoDocumento

2.4.2.1.3.57 TrPerfilUsuario

trewa.bd.trapi.tpo.TrPerfilUsuario

Clase que representa los perfiles de usuario que pueden existir.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPERFILUSU	TpoPK	Identificador del perfil de usuario.
→ NOMBRE	String	Nombre del perfil de usuario.
→ DESCRIPCION	String	Descripción del perfil de usuario.
→ STMA	String	Sistema del perfil de usuario.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPERFILUSU	NUMÉRICO	Identificador del perfil de usuario.
→ CAMPO_NOMBRE	CADENA(25)	Nombre del perfil de usuario.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del perfil de usuario.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema del perfil de usuario.

Métodos para aplicación de los filtros

→ obtenerPerfilUsuario

2.4.2.1.3.58 TrPlantilla

trewa.bd.trapi.tpo.TrPlantilla

Clase que representa las plantillas usadas en la generación de documentos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPLANTILLA	TpoPK	Identificador de la plantilla.
→ NOMBRE	String	Nombre de la plantilla.
→ DESCRIPCION	String	Descripción de la plantilla.
→ NOMBINFORME	String	Nombre del informa de la plantilla.
→ STMA	TrSistema	Sistema de la plantilla.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPLANTILLA	NUMÉRICO	Identificador de la plantilla
→ CAMPO_NOMBRE	CADENA(15)	Nombre de la plantilla
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la plantilla
→ CAMPO_NOMBINFORME	CADENA(15)	Nombre del informe de la plantilla
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema de la plantilla

Métodos para aplicación de los filtros

→ obtenerPlantilla

2.4.2.1.3.59TrPlantillaProcedimiento

trewa.bd.trapi.tpo.TrPlantillaProcedimiento

Clase que representa las posibles plantillas del procedimiento necesarias para comenzar un expediente o que servirán a lo largo de la vida de su tramitación.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFPLANTPROC	TpoPK	Identificador de la plantilla del procedimiento.
→ REFDEFPROC	TpoPK	Identificador de la definición del procedimiento.
→ NOMBRE	String	Nombre de la plantilla del procedimiento.
→ DESCRIPCION	String	Descripción de la plantilla del procedimiento.
→ FICHERO	byte[]	Fichero de la plantilla del procedimiento.
→ NOMBREFICHERO	String	Nombre del fichero físico de la plantilla.
→ FORMATO	String	Tipo de formato del fichero (tipo mime).

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPLANTPROC	NUMÉRICO	Identificador de la plantilla del procedimiento.
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento.
→ CAMPO_NOMBRE	CADENA(100)	Nombre de la plantilla del procedimiento.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la plantilla del procedimiento.
→ CAMPO_NOMBREFICHERO	CADENA(64)	Nombre del fichero físico de la plantilla.
→ CAMPO_FORMATO	CADENA(128)	Tipo de formato del fichero (tipo mime).

Métodos para aplicación de los filtros

→ obtenerPlantillaProcedimiento

2.4.2.1.3.60TrProvincia

trewa.bd.trapi.tpo.TrProvincia

Clase que representa los datos de una provincia.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODPROVINCIA	String	Código de la provincia.
→ NOMBRE	String	Nombre de la provincia.
→ DESCRIPCION	String	Descripción de la provincia.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODPROVINCIA	CADENA(2)	Código de la provincia.
→ CAMPO_NOMBRE	CADENA(32)	Nombre de la provincia.
→ CAMPO_DESCRIPCION	CADENA(64)	Descripción de la provincia.

Métodos para aplicación de los filtros

→ obtenerProvincia

2.4.2.1.3.61 TrPtoTrabOrganismo

trewa.bd.trapi.tpo.TrPtoTrabOrganismo

Clase que representa los distintos puestos de trabajo existentes en cada organismo.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ ORGANISMO	TrOrganismo	Organismo.
→ PUESTOTRABAJO	TrPuestoTrabajo	Puesto de trabajo.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODPTOTRAB	CADENA(10)	Código del puesto de trabajo.
→ CAMPO_NOMBREPTOTRAB	CADENA(32)	Nombre del puesto de trabajo.
→ CAMPO_DESCRIPCIONPTOTRAB	CADENA(64)	Descripción del puesto de trabajo.
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo.
→ CAMPO_CODMUNICIPIO	CADENA(3)	Código del municipio.
→ CAMPO_CODPROVINCIA	CADENA(2)	Código de la provincia.
→ CAMPO_TIPOVIA	CADENA(10)	Código del tipo de vía.
→ CAMPO_CODORG	CADENA(16)	Código del organismo
→ CAMPO_NOMBRE	CADENA(32)	Nombre del organismo.
→ CAMPO_DESCRIPCION	CADENA(200)	Descripción del organismo.
→ CAMPO_TIPO	CADENA(2)	Tipo de organismo. → " DC " – Delegación Provincial → " SC " – Servicios Centrales → " CE " – Centro Adscrito → " EE " – Empresa Externa
→ CAMPO_NOMBREVIA	CADENA(64)	Nombre de la vía pública del domicilio.
→ CAMPO_NUMERO	NUMÉRICO	Número del domicilio.
→ CAMPO_LETRA	CADENA(2)	Letra del domicilio.
→ CAMPO_ESCALERA	CADENA(2)	Escalera del domicilio.
→ CAMPO_PISO	NUMÉRICO	Piso del domicilio.
→ CAMPO_PUERTA	CADENA(2)	Puerta del domicilio.
→ CAMPO_CODPOSTAL	NUMÉRICO	Código postal del domicilio.
→ CAMPO_TELEFONO	CADENA(25)	Número(s) de teléfono de contacto.
→ CAMPO_TLFMOVIL	CADENA(25)	Número(s) de teléfono móvil.
→ CAMPO_FAX	CADENA(25)	Número(s) de Fax.
→ CAMPO_EMAIL	CADENA(64)	Correo electrónico.
→ CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA.
→ CAMPO_IDARIES	CADENA(5)	Identificador de aries.
→ CAMPO_IDENTIFICADOR	CADENA(15)	Número de identificación (CIF).
→ CAMPO_DIGCONTROL	CADENA(1)	Dígito de control para el número de identificación.

→ CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia del organismo.
→ CAMPO_FECHAFINIVIG	DATE	Fecha de fin de vigencia del organismo
→ CAMPO_REFORGPADRE	NUMÉRICO	Identificador del organismo superior al que pertenece.
→ CAMPO_REFTIPOORG	NUMÉRICO	Identificador del tipo de organismo.
→ CAMPO_REFTIPOORGZ	NUMÉRICO	Identificador del tipo de organización.
→ CAMPO_ORDENPRELACION	CADENA(15)	Orden de relación del organismo

Métodos para aplicación de los filtros

→ obtenerPtoTrabOrganismo

2.4.2.1.3.62TrPuestoTrabajo

trewa.bd.trapi.tpo.TrPuestoTrabajo

Clase que representa los distintos puesto de trabajo.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODPTOTRAB	String	Identificador o código del puesto de trabajo.
→ NOMBRE	String	Nombre del puesto de trabajo.
→ DESCRIPCION	String	Descripción del puesto de trabajo.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODPTOTRAB	CADENA(10)	Código del puesto de trabajo.
→ CAMPO_NOMBRE	CADENA(32)	Nombre del puesto de trabajo.
→ CAMPO_DESCRIPCION	CADENA(64)	Descripción del puesto de trabajo.

Métodos para aplicación de los filtros

→ obtenerPuestoTrabajo

2.4.2.1.3.63TrRazonInteres

trewa.bd.trapi.tpo.TrRazonInteres

Clase que representa las razones de interés de los interesados.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFRAZONINT	TpoPK	Identificador de la razón de interés.
→ ABREVIATURA	String	Abreviatura de la razón de interés.
→ DESCRIPCION	String	Descripción de la razón de interés.
→ OBSOLETO	String	Indica si la razón de interés está obsoleta. → " S " – Sí → " N " – No
→ CODWANDA	String	Valor en w@ndA de la razón de interés

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFRAZONINT	NUMÉRICO	Identificador de la razón de interés.
→ CAMPO_ABREVIATUR A	CADENA(10)	Abreviatura de la razón de interés.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción de la razón de interés.
→ CAMPO_OBSOLETO	CADENA(1)	Indica si la razón de interés está obsoleta. → " S " – Sí → " N " – No
→ CAMPO_CODWANDA	CADENA(10)	Valor en w@ndA de la razón de interés

Métodos para aplicación de los filtros

→ obtenerRazonInteres

2.4.2.1.3.64TrRelacion

trewa.bd.trapi.tpo.TrRelacion

Clase que representa las relaciones que se establecen entre fases, transiciones y tipos de documentos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFRELACION	TpoPK	Identificador de la relación
→ FASEA	TrFase	Fase "A".
→ FASEB	TrFase	Fase "B".
→ TRANSICION	TrTransicion	Transición.
→ TIPODOC	TrTipoDocumento	Tipo de documento.
→ TIPORELA	TrTipoRelacion	Tipo de relación.
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ DESCRIPCION	String	Descripción de la relación.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFRELACION	NUMÉRICO	Identificador de la relación.
→ CAMPO_REFFASEA	NUMÉRICO	Identificador de la fase "A".
→ CAMPO_REFFASEB	NUMÉRICO	Identificador de la fase "B".
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición.
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento.
→ CAMPO_REFTIPORELA	NUMÉRICO	Identificador del tipo de relación.
→ CAMPO_REFDEFPROC	NUMÉRICO	Definición del procedimiento.
→ CAMPO_DESCRIPCION	CADENA(200)	Descripción de la relación.

Métodos para aplicación de los filtros

→ obtenerRelacion

2.4.2.1.3.65TrSistema

trewa.bd.trapi.tpo.TrSistema

Clase que representa la información de un sistema.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoría
→ REFSTMA	TpoPK	Identificador del sistema.
→ CODSTMA	String	Código del sistema.
→ DESCRIPCION	String	Descripción del sistema.
→ CONEXIONBUS	String	Indica si se tiene conexión al bus. → " S " – Sí → " N " – No
→ COMPONENTEINF	TrComponente	Componente a informar.
→ COMPONENTEAVI	TrComponente	Componente del avisador.
→ COMPONENTEGUAR	TrComponente	Componente que guarda los documentos.
→ COMPONENTEARC	TrComponente	Componente que archiva.
→ COMPONENTEREG	TrComponente	Componente que registra los documentos.
→ COMPONENTEFIRMA	TrComponente	Componente que firma los documentos.

→ **COMPONENTENOTI** TrComponente Componente que notifica los documentos.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
→ CAMPO_CODSTMA	CADENA(10)	Código del sistema
→ CAMPO_DESCRIPCION	CADENA(128)	Descripción del sistema
→ CAMPO_CONEXIONBUS	CADENA(1)	Indica si se tiene conexión al bus. → "S" – Sí → "N" – No
→ CAMPO_REFCOMPINF	NUMÉRICO	Identificador del componente a informar.
→ CAMPO_REFCOMP AVI	NUMÉRICO	Identificador del componente del avisador.
→ CAMPO_REFCOMP GUA	NUMÉRICO	Identificador del componente que guarda los documentos.
→ CAMPO_REFCOMP ARCH	NUMÉRICO	Identificador del componente que archiva.
→ CAMPO_REFCOMP REG	NUMÉRICO	Identificador del componente que registra los documentos.
→ CAMPO_REFCOMP FIR	NUMÉRICO	Identificador del componente que firma los documentos.
→ CAMPO_REFCOMP NOTI	NUMÉRICO	Identificador del componente que firma los documentos.
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREACION	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICACION	DATE	Fecha de modificación del registro

Métodos para aplicación de los filtros → obtenerSistema

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.2.1.3.66 TrTextoDisposicionFirma

trewa.bd.trapi.tpo.TrTextoDisposicionFirma

Clase que representa los textos para las firmas.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTEXTODISP	TpoPK	Identificador del texto de la disposición.
→ DESCRIPCION	String	Descripción de la disposición.
→ TIPO	String	Tipo de disposición → " D " – Decreto → " T " – Otros → " O " – Orden

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTEXTODISP	NUMÉRICO	Identificador del texto de la disposición.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción de la disposición.
→ CAMPO_TIPO	CADENA(1)	Tipo de disposición → " D " – Decreto → " T " – Otros → " O " – Orden

Métodos para aplicación de los filtros

→ obtenerTextoDisposicionFirma

2.4.2.1.3.67 TrTipoActo

trewa.bd.trapi.tpo.TrTipoActo

Clase que representa los posibles tipos de actos administrativos que pueden existir en un procedimiento administrativo.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
------------------------	--------------------	---------------------------

→ REFTIPOACTO	TpoPK	Identificador del tipo de acto.
→ ABREVIATURA	String	Abreviatura del tipo de acto.
→ DESCRIPCION	String	Descripción del tipo de acto.
→ STMA	TrSistema	Sistema del tipo de acto.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto.
→ CAMPO_ABREVIATURA	CADENA(10)	Abreviatura del tipo de acto.
→ CAMPO_DESCRIPCION	CADENA(200)	Descripción del tipo de acto.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema del tipo de acto.

Métodos para aplicación de los filtros

→ obtenerTipoActo

2.4.2.1.3.68TrTipoComponente

trewa.bd.trapi.tpo.TrTipoComponente

Clase que representa los distintos tipos de componentes que pueden existir en el ámbito w@ndA.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOCOMP	TpoPK	Identificador del tipo de componente.
→ ABREVIATURA	String	Abreviatura del tipo de componente.
→ DESCRIPCION	String	Descripción del tipo de componente.
→ CODWANDA	Integer	Valor en w@ndA del tipo de componente.
→ OBSOLETO	String	Indica si el tipo de componente está obsoleto. → " S " – Sí → " N " – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
---------------------	--------------------	--

→ CAMPO_REFTIPOCOM P	NUMÉRICO	Identificador del tipo de componente
→ CAMPO_ABREVIATURA A	CADENA(10)	Abreviatura del tipo de componente
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción del tipo de componente
→ CAMPO_CODWANDA	NUMÉRICO	Valor en w@ndA del tipo de componente
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de componente está obsoleto. → "S" – Sí → "N" – No

Métodos para aplicación de los filtros

→ obtenerTipoComponente

2.4.2.1.3.69TrTipoContacto

trewa.bd.trapi.tpo.TrTipoContacto

Clase que representa los distintos tipos de contacto que pueden existir entre ciudadanos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOCONT	TpoPK	Identificador del tipo de contacto.
→ ABREVIATURA	String	Abreviatura del tipo de contacto.
→ DESCRIPCION	String	Descripción del tipo de contacto.
→ CODWANDA	String	Valor en w@ndA del tipo de contacto
→ OBSOLETO	String	Indica si el tipo de contacto está obsoleto. → "S" – Sí → "N" – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOCONT	NUMÉRICO	Identificador del tipo de contacto
→ CAMPO_ABREVIATURA A	CADENA(10)	Abreviatura del tipo de contacto
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción del tipo de contacto
→ CAMPO_CODWANDA	NUMÉRICO	Valor en w@ndA del tipo de contacto
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de contacto está obsoleto. → "S" – Sí → "N" – No

Métodos para aplicación de los filtros

→ obtenerTipoContacto

2.4.2.1.3.70TrTipoDocumento

trewa.bd.trapi.tpo.TrTipoDocumento

Clase que representa los tipos de documentos que se pueden dar en cualquier procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPODOC	TrpoPK	Identificador del tipo de documento.
→ ETIQUETA	String	Etiqueta del tipo de documento.
→ NOMBRE	String	Nombre del tipo de documento.
→ DESCRIPCION	String	Descripción del tipo de documento.
→ ENTRADASALIDA	String	Indica si el documento es de entrada o salida. → "E" – Entrada → "S" – Salida → "ES" – Entrada/Salida
→ INCGEN	String	Indica si el documento es generado o incorporado. → "I" – Incorporado → "G" – Generado
→ MULTIPLE	String	Indica si el tipo de documento representa a documentos múltiples. → "S" – Sí → "N" – No
→ TEXTOAUXILIAR	String	Texto auxiliar.
→ FECHAFIRMA	String	Indica si se muestra la fecha en el pie de firma. → "S" – Sí → "N" – No
→ STMA	TrSistema	Sistema.
→ PLANTILLA	TrPlantilla	Plantilla.
→ INFORMAR	String	Indica si se informa al bus de los documento de este tipo.

→ ARCHIVABLE	String	→ "S" – Sí → "N" – No Indica si se permite archivar los documentos de este tipo.
→ VERSIONABLE	String	→ "S" – Sí → "N" – No Indica si se versionan los documentos de este tipo.
→ REUTILIZABLE	String	→ "S" – Sí → "N" – No Indica si se permite reutilizar en otros expedientes los documentos de este tipo.
→ MODOGEN	String	→ "S" – Sí → "N" – No Modo de generación. → "R" – Report Server Oracle → "P" – Generación PDF → "O" – Office Bean
→ FUSIONAR	String	Indica si se fusionan las variables al generar. → "S" – Sí → "N" – No
→ FIRMADIGI	String	Indica si se puede firmar digitalmente. → "S" – Sí → "N" – No
→ TIPOFIRMA	String	Tipo de firma para el documento. → "C" – Cascada → "P" – Paralelo
→ REGISTRABLE	String	Indica si se puede registrar el documento. → "S" – Sí → "N" – No
→ NOTIFICABLE	String	Indica si se notifica. → "S" – Sí → "N" – No
→ OBSOLETO	String	Indica si el documento está obsoleto. → "S" – Sí → "N" – No
→ EDITABLE	String	Indica si el documento es editable por el tramitador. → "S" – Sí → "N" – No
→ ADMINISTRATIVO	String	Indica si el documento es administrativo o auxiliar. → "S" – Sí → "N" – No
→ CODWANDA	String	Valor en w@ndA del tipo de documento.
→ CLASEDOC	String	Indica de qué clase es el documento. Por ejemplo SCSP o GENERAL.
→ VALIDA_CONSULTA	String	Indica si la consulta de plantilla múltiple es valida o no. → "S" – Si → "N" – No
→ CONSULTA_PLAN T_MULT	TrConsPlantMult	Consulta que se aplicará para calcular la plantilla múltiple del tipo documento al generar lo.

Campos definidos para filtrados y ordenación



<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento.
→ CAMPO_ETIQUETA	CADENA(11)	Etiqueta del tipo de documento.
→ CAMPO_NOMBRE	CADENA(50)	Nombre del tipo de documento.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del tipo de documento.
→ CAMPO_ENTRADASALIDA	CADENA(2)	Indica si el documento es de entrada o salida. → "E" – Entrada → "S" – Salida → "ES" – Entrada/Salida
→ CAMPO_INCGEN	CADENA(1)	Indica si el documento es generado o incorporado. → "I" – Incorporado → "G" – Generado
→ CAMPO_MULTIPLE	CADENA(1)	Indica si el tipo de documento representa a documentos múltiples. → "S" – Sí → "N" – No
→ CAMPO_TEXTOAUXILIAR	CADENA(50)	Texto auxiliar.
→ CAMPO_FECHAFIRMA	CADENA(1)	Indica si se muestra la fecha en el pie de firma. → "S" – Sí → "N" – No
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema
→ CAMPO_REFPLANTILLA	NUMÉRICO	Identificador de la plantilla
→ CAMPO_INFOMAR	CADENA(1)	Indica si se informa al bus de los documento de este tipo. → "S" – Sí → "N" – No
→ CAMPO_ARCHIVABLE	CADENA(1)	Indica si se permite archivar los documentos de este tipo. → "S" – Sí → "N" – No
→ CAMPO_VERSIONABLE	CADENA(1)	Indica si se versionan los documentos de este tipo. → "S" – Sí → "N" – No
→ CAMPO_REUTILIZABLE	CADENA(1)	Indica si se permite reutilizar en otros expedientes los documentos de este tipo. → "S" – Sí → "N" – No
→ CAMPO_MODALOGEN	CADENA(1)	Modo de generación. → "R" – Report Server Oracle → "P" – Generación PDF → "O" – Office Bean
→ CAMPO_FUSIONAR	CADENA(1)	Indica si se fusionan las variables al generar. → "S" – Sí → "N" – No
→ CAMPO_FIRMADIGI	CADENA(1)	Indica si se puede firmar digitalmente. → "S" – Sí → "N" – No
→ CAMPO_TIPOFIRMA	CADENA(1)	Tipo de firma para el documento. → "C" – Cascada → "P" – Paralelo

→ CAMPO_FORMATO	CADENA(128)	Tipo de formato de la plantilla (tipo mime).
→ CAMPO_NOMBREFICHERO	CADENA(64)	Nombre del fichero de la plantilla.
→ CAMPO_REGISTRABLE	CADENA(1)	Indica si se puede registrar el documento. → "S" – Sí → "N" – No
→ CAMPO_NOTIFICABLE	CADENA(1)	Indica si se notifica. → "S" – Sí → "N" – No
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el documento está obsoleto. → "S" – Sí → "N" – No
→ CAMPO_EDITABLE	CADENA(1)	Indica si el documento es editable. → "S" – Sí → "N" – No
→ CAMPO_ADMINISTRATIVO	CADENA(1)	Indica si el documento es administrativo o auxiliar. → "S" – Sí → "N" – No
→ CAMPO_CODWANDA	CADENA(30)	Valor en w@ndA del tipo de documento.
→ CAMPO_CLASEDOC	CADENA(20)	Indica de qué clase es el documento. Por ejemplo SCSP o GENERAL.
→ CAMPO_REFCONSULTA	NUMÉRICO	Identificador de la consulta para la generación de la plantilla múltiple del tipo de documento
→ CAMPO_CONSVALIDA	CADENA(1)	Indica si la consulta de la plantilla múltiple del tipo de documento es válida. → "S" – Sí → "N" – No

Métodos para aplicación de los filtros

→ obtenerTipoDocumento

2.4.2.1.3.71 TrTipoExpediente

trewa.bd.trapi.tpo.TrTipoExpediente

Clase que representa un tipo de expediente.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoría
→ REFTIPOEXP	TpoPK	Identificador del tipo de expediente.
→ ABREVIATURA	String	Abreviatura del tipo de expediente.
→ DESCRIPCION	String	Descripción del tipo de expediente.
→ VIGENTE	String	Indica si el tipo de expediente está vigente. → "S" – Sí → "N" – No
→ STMA	TrSistema	Sistema del tipo de expediente.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOEXP	NUMÉRICO	Identificador del tipo de expediente.
→ CAMPO_ABREVIATUR A	CADENA(10)	Abreviatura del tipo de expediente.
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción del tipo de expediente.
→ CAMPO_VIGENTE	CADENA(1)	Indica si el tipo de expediente está vigente. → "S" – Sí → "N" – No
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema del tipo de expediente.
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREAD O	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFI CA	DATE	Fecha de modificación del registro

Métodos para aplicación de los filtros y dónde se usa

- explorarExpedientesSimpl
- obtenerExpedientes
- obtenerExpedientesSimple
- obtenerExpsRelacionInteresado
- obtenerTipoExpAndTipoEvVersionDefProc
- obtenerTipoExpedienteSimple
- obtenerVersionDefProcedimiento

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.2.1.3.72TrTipoidentificador

trewa.bd.trapi.tpo.TrTipoidentificador

Clase que representa los distintos tipos de identificadores (DNI, NIF...).

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODTIPOIDENT	String	Código del tipo de identificador.
→ DESCRIPCION	String	Descripción del tipo de identificador.
→ TIPOIDENT	String	Indica si se trata de una persona física o jurídica. → "F" – Física → "J" – Jurídica
→ CODWANDA	String	Valor en w@ndA del tipo de identificador
→ OBSOLETO	String	Indica si el tipo de identificador está obsoleto. → "S" – Sí → "N" – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODTIPOIDENT T	NUMÉRICO	Código del tipo de identificador.
→ CAMPO_DESCRIPCION	CADENA(10)	Descripción del tipo de identificador.
→ CAMPO_TIPOIDENT	CADENA(1)	Indica si se trata de una persona física o jurídica. → "F" – Física → "J" – Jurídica
→ CAMPO_CODWANDA	CADENA(30)	Valor en w@ndA del tipo de identificador
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de identificador está obsoleto. → "S" – Sí → "N" – No

Métodos para aplicación de los filtros

→ obtenerTipoidentificador

2.4.2.1.3.73TrTipoIndicacion

trewa.bd.trapi.tpo.TrTipoIndicacion

Clase que representa los tipos de datos variables sobre la ficha descriptiva del procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOIND	TpoPK	Identificador del tipo de indicación de la ficha.
→ ABREVIATURA	String	Abreviatura del tipo de indicación de la ficha descriptiva del procedimiento.
→ DESCRIPCION	String	Descripción del tipo de indicación de la ficha descriptiva

- **CODWANDA** String del procedimiento.
Valor en w@ndA del tipo de indicación de la ficha descriptiva del procedimiento.
- **OBSOLETO** String Indica si el tipo de indicación está obsoleto.
→ "**S**" – Sí
→ "**N**" – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOIND	NUMÉRICO	Identificador del tipo de indicación de la ficha.
→ CAMPO_ABREVIATUR A	CADENA(10)	Abreviatura del tipo de indicación de la ficha descriptiva del procedimiento.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción del tipo de indicación de la ficha descriptiva del procedimiento.
→ CAMPO_CODWANDA	CADENA(30)	Valor en w@ndA del tipo de indicación de la ficha descriptiva del procedimiento.
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de indicación está obsoleto. → " S " – Sí → " N " – No

Métodos para aplicación de los filtros

- obtenerIndicacionFicha
- obtenerTipoIndicacion

2.4.2.1.3.74TrTipoNormativa

trewa.bd.trapi.tpo.TrTipoNormativa

Clase que representa la información de los distintos tipos de normativas.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPONORM	TpoPK	Identificador del tipo de normativa.
→ ABREVIATURA	String	Abreviatura del tipo de normativa.
→ DESCRIPCION	String	Descripción del tipo de normativa.
→ CODWANDA	String	Valor en w@ndA del tipo de normativa.
→ OBSOLETO	String	Indica si el tipo de normativa está obsoleto. → " S " – Sí → " N " – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
---------------------	--------------------	--

- **CAMPO_REFTIPONOR** NUMÉRICO Identificador del tipo de normativa.
- M**
- **CAMPO_ABREVIATUR** CADENA(10) Abreviatura del tipo de normativa.
- A**
- **CAMPO_DESCRIPCION** CADENA(100) Descripción del tipo de normativa.
- **CAMPO_CODWANDA** CADENA(30) Valor en w@ndA del tipo de normativa.
- **CAMPO_OBSOLETO** CADENA(1) Indica si el tipo de normativa está obsoleto.
- "S" – Sí
- "N" – No

Métodos para aplicación de los filtros

→ obtenerTipoNormativa

2.4.2.1.3.75TrTipoOrganismo

trewa.bd.trapi.tpo.TrTipoOrganismo

Clase que representa los distintos tipos de organismos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOORG	TpoPK	Identificador del tipo de organismo.
→ ABREVIATURA	String	Abreviatura del tipo de organismo.
→ DESCRIPCION	String	Descripción del tipo de organismo.
→ CODWANDA	String	Valor en w@ndA del tipo de organismo
→ OBSOLETO	String	Indica si el tipo de organismo está obsoleto.
		→ "S" – Sí
		→ "N" – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOORG	NUMÉRICO	Identificador del tipo de organismo.
→ CAMPO_ABREVIATUR	CADENA(10)	Abreviatura del tipo de organismo.
A		
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción del tipo de organismo.
→ CAMPO_CODWANDA	CADENA(1)	Valor en w@ndA del tipo de organismo.
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de organismo está obsoleto.
		→ "S" – Sí
		→ "N" – No

Métodos para aplicación de los filtros

→ obtenerTipoOrganismo

2.4.2.1.3.76TrOrganismoProcedimiento

trewa.bd.trapi.tpo.TrOrganismoProcedimiento

Clase que representa la relación entre un organismo y un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFORGP	TpoPK	Identificador de la relación entre el organismo y el procedimiento
→ ORGANISMO	TrOrganismo	Organismo
→ REFDEFPROC	TpoPK	Identificador del organismo
→ TIPOASOCIA	String	Tipo de asociación entre el organismo y el procedimiento
→ OBSERVACIONES	String	Observaciones
→ FECHAINIVIG	Timestamp	Fecha de inicio de vigencia
→ FECHAFINIVIG	Timestamp	Fecha de fin de vigencia

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFORGP	NUMÉRICO	Identificador de la relación entre el organismo y el procedimiento
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador del procedimiento
→ CAMPO_TIPOASOCIA	CADENA	Tipo de asociación entre el organismo y el procedimiento
→ CAMPO_OBSERVACIONES	CADENA	Observaciones
→ CAMPO_FECHAINIVIG	DATE	Fecha de inicio de vigencia
→ CAMPO_FECHAFINIVIG	DATE	Fecha de fin de vigencia
→ CAMPO_REFORGANISMO	NUMÉRICO	Identificador del organismo
→ CAMPO_CODMUNICIPIO	CADENA(3)	Código del municipio
→ CAMPO_CODPROVINCIA	CADENA(2)	Código de la provincia
→ CAMPO_TIPOVIA	CADENA(10)	Tipo de vía
→ CAMPO_CODORG	CADENA(16)	Código del organismo
→ CAMPO_NOMBREORG	CADENA(32)	Nombre del organismo
→ CAMPO_DESCRIPCIONORG	CADENA(200)	Descripción del organismo
→ CAMPO_TIPO	CADENA(1)	Tipo de organismo → 'DP' – Delegación Provincial → 'SC' – Servicios Centrales → 'CE' – Centro Adscrito → 'EE' – Empresa Externa
→ CAMPO_NOMBREVIA	CADENA(64)	Nombre de la vía
→ CAMPO_NUMERO	NUMÉRICO	Número del domicilio
→ CAMPO_LETRA	CADENA(2)	Letra del domicilio
→ CAMPO_ESCALERA	CADENA(2)	Escalera del domicilio
→ CAMPO_PISO	NUMERICO	Piso del domicilio

→ CAMPO_PUERTA	CADENA(2)	Puerta del domicilio
→ CAMPO_CODPOSTAL	NUMERICO	Código postal del domicilio
→ CAMPO_TELEFONO	CADENA(25)	Número(s) de teléfono de contacto
→ CAMPO_TLFMOVIL	CADENA(25)	Número(s) de teléfono móvil
→ CAMPO_FAX	CADENA(25)	Número(s) de Fax
→ CAMPO_EMAIL	CADENA(64)	Correo electrónico
→ CAMPO_CIWA	CADENA(15)	Código de identificación w@ndA
→ CAMPO_IDARIES	CADENA(5)	Identificador de aries
→ CAMPO_IDENTIFICADOR	CADENA(15)	Número de identificador
→ CAMPO_DIGCONTROL	CADENA(1)	Dígito de control
→ CAMPO_FECHAINIVIGORG	DATE	Fecha de inicio de vigencia del organismo
→ CAMPO_FECHAFINVIGOR	DATE	Fecha de fin de vigencia del organismo
G		
→ CAMPO_REFORGPADRE	NUMÉRICO	Identificador del organismo superior al que pertenece.
→ CAMPO_REFTIPOORG	NUMÉRICO	Identificador del tipo de organismo.
→ CAMPO_REFTIPOORGZ	NUMÉRICO	Identificador del tipo de organización.
→ CAMPO_ORDENPRELACIO	CADENA(15)	Orden de relación del organismo
N		

Métodos para aplicación de los filtros

→ obtenerOrganismosProcedimiento

2.4.2.1.3.77TrTipoOrganizacion

trewa.bd.trapi.tpo.TrTipoOrganizacion

Clase que representa los distintos tipos de organismos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODTIPOORGZ	String	Código del tipo de organización.
→ DESCRIPCION	String	Descripción del tipo de organización.
→ VIGENTE	String	Indica si el tipo de organización está vigente. → "S" – Sí → "N" – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODTIPOORGZ	CADENA(1)	Código del tipo de organización.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción del tipo de organización.
→ CAMPO_VIGENTE	CADENA(1)	Indica si el tipo de organización está vigente. → "S" – Sí

→ "N" – No

Métodos para aplicación de los filtros

→ obtenerTipoOrganizacion

2.4.2.1.3.78TrTipoParrafo

trewa.bd.trapi.tpo.TrTipoParrafo

Clase que representa un tipo de párrafo para los documentos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOPARR	TpoPK	Identificador del tipo de párrafo.
→ ABREVIATURA	String	Abreviatura del tipo de párrafo.
→ DESCRIPCION	String	Descripción del tipo de párrafo.
→ PARRAFO	String	Contenido del tipo de párrafo.
→ ALINEACION	String	Alineación del tipo de párrafo. → "I" – Izquierda → "C" – Centrado → "D" – Derecha → "J" – Justificado
→ ETIQUETA	String	Etiqueta del tipo de párrafo.
→ ESTILO	String	Estilo del tipo del párrafo. → "NORMAL" – Normal → "NEGRIT" – Negrita → "CURSIV" – Cursiva → "SUBRAY" – Subrayado → "NE-CU" – Negrita y Cursiva → "NE-SU" – Negrita y Subrayado → "CU-SU" – Cursiva y Subrayado → "NE-CU-SU" – Negrita, Cursiva y Subrayado
→ ESTILOETIQ	String	Estilo de la etiqueta del tipo de párrafo. → "NORMAL" – Normal → "NEGRIT" – Negrita → "CURSIV" – Cursiva → "SUBRAY" – Subrayado → "NE-CU" – Negrita y Cursiva → "NE-SU" – Negrita y Subrayado → "CU-SU" – Cursiva y Subrayado → "NE-CU-SU" – Negrita, Cursiva y Subrayado
→ UBICACIÓN	String	Ubicación del tipo de párrafo. → "A" – Cabecera → "C" – Cuerpo

→ EDITABLE	String	→ "P" – Pie → "O" – Otro Indica si el tipo de párrafo es editable. → "S" – Sí → "N" – No
→ IMAGEN	byte[]	Imagen del tipo de párrafo
→ STMA	TrSistema	Sistema.
→ NOMBREFICHERO	String	Nombre del fichero de la imagen física.
→ FORMATO	String	Tipo de formato de la imagen (tipo mime).

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOPARR	NUMÉRICO	Identificador del tipo de párrafo.
→ CAMPO_ABREVIATUR A	CADENA(15)	Abreviatura del tipo de párrafo.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del tipo de párrafo.
→ CAMPO_PARRAFO	CADENA(4000)	Contenido del tipo de párrafo.
→ CAMPO_ALINEACION	CADENA(1)	Alineación del tipo de párrafo. → "I" – Izquierda → "C" – Centrado → "D" – Derecha → "J" – Justificado
→ CAMPO_ETIQUETA	CADENA(15)	Etiqueta del tipo de párrafo.
→ CAMPO_ESTILO	CADENA(8)	Estilo del tipo del párrafo. → "NORMAL" – Normal → "NEGRIT" – Negrita → "CURSIV" – Cursiva → "SUBRAY" – Subrayado → "NE-CU" – Negrita y Cursiva → "NE-SU" – Negrita y Subrayado → "CU-SU" – Cursiva y Subrayado → "NE-CU-SU" – Negrita, Cursiva y Subrayado
→ CAMPO_ESTILOETIQ	CADENA(8)	Estilo de la etiqueta del tipo de párrafo. → "NORMAL" – Normal → "NEGRIT" – Negrita → "CURSIV" – Cursiva → "SUBRAY" – Subrayado → "NE-CU" – Negrita y Cursiva → "NE-SU" – Negrita y Subrayado → "CU-SU" – Cursiva y Subrayado → "NE-CU-SU" – Negrita, Cursiva y Subrayado
→ CAMPO_UBICACION	CADENA(1)	Ubicación del tipo de párrafo. → "A" – Cabecera → "C" – Cuerpo → "P" – Pie → "O" – Otro

- **CAMPO_EDITABLE** CADENA(1) Indica si el tipo de párrafo es editable.
→ "S" – Sí
→ "N" – No
- **CAMPO_REFSTMA** NUMÉRICO Identificador del sistema.
- **CAMPO_NOMBREFICHERO** CADENA(64) Nombre del fichero de la imagen física.
- **CAMPO_FORMATO** CADENA(128) Tipo de formato de la imagen (tipo mime).

Métodos para aplicación de los filtros → obtenerTipoParrafo

2.4.2.1.3.79TrTipoPublicacion

trewa.bd.trapi.tpo.TrTipoPublicacion

Clase que representa la información de los distintos tipos de publicación de las normativas.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPOPUB	TpoPK	Identificador del tipo de publicación de la normativa
→ ABREVIATURA	String	Abreviatura del tipo de publicación.
→ DESCRIPCION	String	Descripción del tipo de publicación.
→ CODWANDA	Integer	Valor en w@ndA del tipo de publicación
→ OBSOLETO	String	Indica si el tipo de publicación está obsoleto. → "S" – Sí → "N" – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPOPUB	NUMÉRICO	Identificador del tipo de publicación de la normativa
→ CAMPO_ABREVIATURA	CADENA(10)	Abreviatura del tipo de publicación.
→ CAMPO_DESCRIPCION	CADENA(100)	Descripción del tipo de publicación.
→ CAMPO_CODWANDA	NUMÉRICO	Valor en w@ndA del tipo de publicación
→ CAMPO_OBSOLETO	CADENA(50)	Indica si el tipo de publicación está obsoleto. → "S" – Sí → "N" – No

Métodos para aplicación de los filtros → obtenerTipoPublicacion

2.4.2.1.3.80 TrTipoRelacion

trewa.bd.trapi.tpo.TrTipoRelacion

Clase que representa las posibles relaciones a establecer.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPORELA	TpoPK	Identificador del tipo de relación.
→ NOMBRE	String	Nombre del tipo de relación.
→ DESCRIPCION	String	Descripción del tipo de relación.
→ STMA	TrSistema	Sistema del tipo de relación.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPORELA	NUMÉRICO	Identificador del tipo de relación.
→ CAMPO_NOMBRE	CADENA(25)	Nombre del tipo de relación.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción del tipo de relación.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema del tipo de relación.

Métodos para aplicación de los filtros

→ obtenerTipoRelacion

2.4.2.1.3.81 TrTipoVia

trewa.bd.trapi.tpo.TrTipoVia

Clase que representa los distintos tipos de vía.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ CODTIPOVIA	String	Código del tipo de vía.
→ DESCRIPCION	String	Descripción del tipo de vía.
→ CODWANDA	String	Valor en w@ndA del tipo de vía
→ OBSOLETO	String	Indica si el tipo de vía está obsoleto. → "S" – Sí → "N" – No

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODTIPOVIA	CADENA(10)	Código del tipo de vía.
→ CAMPO_DESCRIPCION	CADENA(32)	Descripción del tipo de vía.
→ CAMPO_CODWANDA	CADENA(15)	Valor en w@ndA del tipo de vía
→ CAMPO_OBSOLETO	CADENA(1)	Indica si el tipo de vía está obsoleto. → "S" – Sí → "N" – No

Métodos para aplicación de los filtros

→ obtenerTipoVia

2.4.2.1.3.82TrTransicion

trewa.bd.trapi.tpo.TrTransicion

Clase que representa una transición

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTRANSICION	TpoPK	Identificador de la transición
→ ETIQUETA	String	Etiqueta de la transición
→ ORDEN	Integer	Orden entre las posibles transiciones que se pueden dar desde una determinada fase.
→ DESCRIPCION	String	Descripción de la transición.
→ DESCFECHA	String	Descripción de lo que significa la fecha de realización de la transición.
→ TIPO	String	Indica el tipo de la transición. → "FA" – Fase normal → "I" – Fase inicial → "D" – División → "U" – Unión → "F" – Fase final → "ES" – Evento salida, abandona la fase actual. → "EN" – Evento que no abandona la fase actual.
→ VALIDA	String	Indica si la transición es válida → "S" – Sí → "N" – No
→ FASEINI	TrFase	Fase inicial
→ FASEFIN	TrFase	Fase final
→ REFTRANSPADRE	TpoPK	Identificador de la transición padre
→ TIPOACTO	TrTipoActo	Tipo de acto.
→ INFORMAR	String	Indica si se informa al bus → "S" – Sí

→ **ETIQLARGA** String → "N" – No
Etiqueta larga de la transición

2.4.2.1.3.83TrTransicionDefProcedimiento

trewa.bd.trapi.tpo.TrTransicionDefProcedimiento

Clase que representa una transición asociada a un procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFDEFPROC	TpoPK	Identificador de la definición de procedimiento.
→ TRANSICION	TrTransicion	Transición.
→ NUMMAX	Integer	Número máximo de veces por la que se puede pasar por una transición en un procedimiento.
→ UNIDAD	String	Unidad de medida. → "D" – Días → "M" – Meses → "A" – Años
→ NUMUNIDADES	String	Número unidades: días, meses o años.
→ DESCFECHALIM	String	Descripción del significado de la fecha límite de estancia en la fase.
→ REFTRANPROV	TpoPK	Identificador de la transición provocada por la fecha límite

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición de procedimiento
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición
→ CAMPO_NUMMAX	NUMÉRICO	Número máximo de veces por la que se puede pasar por una transición en un procedimiento
→ CAMPO_UNIDAD	CADENA(1)	Unidad de medida. → "D" – Días → "M" – Meses → "A" – Años
→ CAMPO_NUMUNIDADE	NUMÉRICO	Número unidades: días, meses o años.



S		
→ CAMPO_DESCFECHALIM	CADENA(50)	Descripción del significado de la fecha límite de estancia en la fase
→ CAMPO_REFTRANPROV	NUMÉRICO	Identificador de la transición provocada por la fecha límite
→ CAMPO_ETIQUETA	CADENA(11)	Etiqueta de la transición
→ CAMPO_ORDEN	NUMÉRICO	Orden entre las posibles transiciones que se pueden dar desde una determinada fase.
→ CAMPO_DESCTRANSICION	CADENA(50)	Descripción de la transición.
→ CAMPO_DESCFECHA	CADENA(50)	Descripción de lo que significa la fecha de realización de la transición
→ CAMPO_TIPO	CADENA(2)	Tipo de la transición. → “ N ” – Normal → “ D ” – División → “ U ” – Unión → “ ES ” – Evento de salida, abandona la fase actual. → “ EN ” – Evento que no provoca la salida de la fase actual.
→ CAMPO_VALIDA	CADENA(1)	Indica si la transición es válida o no. → “ S ” – Sí es válida → “ N ” – No es válida.
→ CAMPO_REFFASEINI	NUMÉRICO	Identificador de la fase inicial de la transición.
→ CAMPO_REFSTMAFAS EINI	NUMÉRICO	Identificador del sistema de la fase inicial de la transición.
→ CAMPO_NOMBREFASE INI	CADENA(50)	Nombre de la fase inicial de la transición.
→ CAMPO_REFFASEFIN	NUMÉRICO	Identificador de la fase final de la transición.
→ CAMPO_REFSTMAFAS EFIN	NUMÉRICO	Identificador del sistema de la fase final de la transición.
→ CAMPO_NOMBREFASE FIN	CADENA(50)	Nombre de la fase final de la transición.
→ CAMPO_REFTRANXTRAN	NUMÉRICO	Identificador de la transición que se divide.
→ CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto.
→ CAMPO_INFORMAR	CADENA(1)	Indica si se informa al bus del cambio de fase. → “ S ” – Sí. → “ N ” – No.
→ CAMPO_ETIQLARGA	CADENA(25)	Etiqueta larga de la transición.

Métodos para aplicación de los filtros

→ obtenerTransicionDefProcedimiento

2.4.2.1.3.84 TrTransicionGr

trewa.bd.trapi.tpo.TrTransicionGr

Clase que representa la información de una transición gráfica.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTRANSGR	TpoPK	Identificador de la transición gráfica.
→ EXTGRINI	TrExtremoTransicionGr	Extremo gráfico inicial.
→ EXTGRFIN	TrExtremoTransicionGr	Extremo gráfico final.
→ TRANSICION	TrTransicion	Transición.
→ DEFPROCGR	TrDefProcedimientoGr	Definición de procedimiento.
→ COLORLINEA	Long	Color de línea de la transición.
→ COLORTEXTO	Long	Color del texto de la etiqueta de la transición.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTRANSGR	NUMÉRICO	Identificador de la transición gráfica.

- **CAMPO_REFEXTGRINI** NUMÉRICO Identificador del extremo gráfico inicial.
- **CAMPO_REFEXTGRFIN** NUMÉRICO Identificador del extremo gráfico final.
- **CAMPO_REFTRANSICION** NUMÉRICO Identificador de la transición.
- **CAMPO_REFDEFPROCGR** NUMÉRICO Identificador de la definición del procedimiento gráfico.
- **CAMPO_COLORLINEA** NUMÉRICO Color de línea de la transición.
- **CAMPO_COLORTEXTO** NUMÉRICO Color del texto de la etiqueta de la transición.

Métodos para aplicación de los filtros

→ obtenerTransicionGr

2.4.2.1.3.85TrTransicionPerfil

trewa.bd.trapi.tpo.TrTransicionPerfil

Clase que representa un perfil de usuario que tiene permisos para realizar una determinada transición.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ TRANDEFPROC	TrTransicionDefProcedimiento	Transición asociada a una definición de procedimiento.
→ REFPERFILUSU	TpoPK	Identificador del perfil de usuario.
→ PERMISO	String	Tipo de permiso. → “ T ” – Tramitar → “ D ” – Deshacer → “ A ” – Ambos

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición de procedimiento.
→ CAMPO_REFTRANSICION	NUMÉRICO	Identificador de la transición.
→ CAMPO_REFPERFILUSU	NUMÉRICO	Identificador del perfil de usuario.
→ CAMPO_PERMISO	CADENA(1)	Tipo de permiso. → “ T ” – Tramitar → “ D ” – Deshacer → “ A ” – Ambos

Métodos para aplicación de los filtros

→ obtenerTransicionPerfil

2.4.2.1.3.86TrUsuario

trewa.bd.trapi.tpo.TrUsuario

Clase que representa la información de un usuario.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ AUDITORIA	TrAuditoria	Datos de auditoría
→ CODUSUARIO	String	Código del usuario.
→ TIPOIDENT	String	Tipo de identificador. → "D" – DNI / NIF → "P" – Pasaporte → "N" – NIE → "C" – CIF

→ IDENTIFICADOR	String	→ "O" – Otros Código de identificación.
→ NOMBRE	String	Nombre del usuario.
→ APELLIDO1	String	Primer apellido del usuario.
→ APELLIDO2	String	Extremo gráfico inicial.
→ SEXO	String	Sexo del usuario. → "F" – Femenino → "M" – Masculino
→ EMAIL	String	Dirección de correo electrónico del usuario.
→ CLAVE	String	Clave del usuario (típicamente encriptada).
→ FECHAALTA	Timestamp	Fecha de alta en el sistema.
→ FECHABAJA	Timestamp	Fecha de baja en el sistema.
→ ANAGRAMAFISCAL	String	Anagrama fiscal del usuario

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_CODUSUARIO	CADENA(10)	Código del usuario.
→ CAMPO_TIPOIDENT	CADENA(1)	Tipo de identificador. → "D" – DNI / NIF → "P" – Pasaporte → "N" – NIE → "C" – CIF → "O" – Otros
→ CAMPO_IDENTIFICADOR	CADENA(15)	Código de identificación.
→ CAMPO_NOMBRE	CADENA(32)	Nombre del usuario.
→ CAMPO_APELLIDO1	CADENA(32)	Primer apellido del usuario.
→ CAMPO_APELLIDO2	CADENA(32)	Extremo gráfico inicial.
→ CAMPO_SEXO	CADENA(1)	Sexo del usuario. → "F" – Femenino → "M" – Masculino
→ CAMPO_EMAIL	CADENA(64)	Dirección de correo electrónico del usuario.
→ CAMPO_CLAVE	CADENA(32)	Clave del usuario (típicamente encriptada).
→ CAMPO_FECHAALTA	DATE	Fecha de alta en el sistema.
→ CAMPO_FECHABAJA	DATE	Fecha de baja en el sistema.
→ CAMPO_ANAGRAMAFISCAL	VARCHAR2(50)	Anagrama fiscal del usuario
→ CAMPO_CREADO	CADENA	Usuario que crea el registro
→ CAMPO_FECHACREACION	DATE	Fecha de creación del registro
→ CAMPO_MODIFICADO	CADENA	Usuario que modifica el registro
→ CAMPO_FECHAMODIFICACION	DATE	Fecha de modificación del registro

Métodos para aplicación de los filtros

→ obtenerUsuario

	<p>Junta de Andalucía Agencia Digital de Andalucía</p>	<p>Trew@ Guía de referencia de JTrApi</p>
---	---	---

Campos definidos para utilidades

<u>Campo</u>	<u>Descripción y restricciones de uso</u>
→ MAX_ROW	Indica el número máximo de filas a devolver. El valor debe de ser numérico y mayor que cero.
→ INIT_ROW	Indica el primer registro a devolver. Útil para realizar paginados combinado con MAX_ROW. El valor debe de ser numérico. El primer registro tiene índice cero.

2.4.2.1.3.87TrUsuarioPerfilUsuario

trewa.bd.trapi.tpo.TrUsuarioPerfilUsuario

Clase que representa un perfil de usuario asignado a un usuario.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
------------------------	--------------------	---------------------------

- **PERFILUSU** TrPerfilUsuario Perfil de usuario.
- **USUARIO** String Código del usuario.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFPERFILUSU	NUMÉRICO	Identificador del perfil de usuario.
→ CAMPO_USUARIO	CADENA(10)	Código del usuario.

Métodos para aplicación de los filtros

→ obtenerUsuarioPerfilUsuario

2.4.2.1.3.88TrVariable

trewa.bd.trapi.tpo.TrVariable

Clase que representa una variable definida en el tramitador para los documentos.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
-----------------	-------------	--------------------

→ REFVARIABLE	TpoPK	Identificador de la variable.
→ NOMBRE	String	Nombre de la variable.
→ DESCRIPCION	String	Descripción de la variable.
→ FUNCION	String	Nombre de la función que calcula la variable.
→ STMA	TrSistema	Sistema.
→ TIPOACTO	TrTipoActo	Tipo de acto.
→ PAQUETE	String	Nombre del paquete que contiene la función almacenada o método java.
→ IMPLEMENTACION	String	Indica el tipo de implementación. → “ F ” – Función PL/SQL → “ J ” – Java

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFVARIABLE	NUMÉRICO	Identificador de la variable.
→ CAMPO_NOMBRE	CADENA(25)	Nombre de la variable.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la variable.
→ CAMPO_FUNCION	CADENA(50)	Nombre de la función que calcula la variable.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema.
→ CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto.
→ CAMPO_PAQUETE	CADENA(100)	Nombre del paquete que contiene la función almacenada o método java.
→ CAMPO_IMPLEMENTACION	CADENA(1)	Indica el tipo de implementación. → “ F ” – Función PL/SQL → “ J ” – Java

Métodos para aplicación de los filtros

→ obtenerVariable

2.4.2.1.3.89TrVariableTipoDocumento

trewa.bd.trapi.tpo.TrVariableTipoDocumento

Clase que representa las variables que en principio se definen para un tipo de documento. Sólo válido para plantillas de Open Office.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ REFTIPODOC	TpoPK	Identificador del tipo de documento.
→ VARIABLE	TrVariable	Variable.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFTIPODOC	NUMÉRICO	Identificador del tipo de documento.
→ CAMPO_REFVARIABLE	NUMÉRICO	Identificador de la variable.
→ CAMPO_NOMBRE	CADENA(25)	Nombre de la variable.
→ CAMPO_DESCRIPCION	CADENA(250)	Descripción de la variable.
→ CAMPO_FUNCION	CADENA(50)	Nombre de la función que calcula la variable.
→ CAMPO_REFSTMA	NUMÉRICO	Identificador del sistema.
→ CAMPO_REFTIPOACTO	NUMÉRICO	Identificador del tipo de acto.
→ CAMPO_PAQUETE	CADENA(100)	Nombre del paquete que contiene la función almacenada o método java.
→ CAMPO_IMPLEMENTACION	CADENA(1)	Indica el tipo de implementación. → "F" – Función PL/SQL → "J" – Java

Métodos para aplicación de los filtros

→ obtenerVariableTipoDocumento

2.4.2.1.3.90TrVersionDefProcedimiento

trewa.bd.trapi.tpo.TrVersionDefProcedimiento

Clase que representa una versión de una definición de procedimiento.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ DEFPROC	TrDefProcedimiento	Definición del procedimiento.
→ TIPOEXP	TrTipoExpediente	Tipo de expediente.
→ FEHAVIGOR	Timestamp	Fecha en la que entra en vigor la definición del procedimiento.
→ DESCRIPCION	String	Descripción de la versión de la definición del procedimiento.

Campos definidos para filtrados y ordenación

<u>Campo</u>	<u>Tipo</u>	<u>Descripción y restricciones de uso</u>
→ CAMPO_REFDEFPROC	NUMÉRICO	Identificador de la definición del procedimiento
→ CAMPO_REFTIPOEXP	NUMÉRICO	Identificador del tipo de expediente
→ CAMPO_FEHAVIGOR	DATE	Fecha en la que entra en vigor la definición del procedimiento.
→ CAMPO_DESCRIPCION	CADENA(50)	Descripción de la versión de la definición del procedimiento.

Métodos para aplicación de los filtros

→ obtenerVersionDefProcedimiento

2.4.2.2 Métodos de la TrAPIADM

2.4.2.2.1 Mantenimiento de Certificados

2.4.2.2.1.1 Insertar certificado

TpoPK insertarCertificado(TrCertificado cert) throws TrException

Inserta un nuevo certificado de @firma. Si todo es correcto devuelve el identificador del nuevo certificado, si algo falla lanza una excepción.

Entradas

TrCertificado cert Certificado a insertar

Salida

TpoPK Identificador del certificado insertado, cero si no se ha insertado

2.4.2.2.1.2 Modificar certificado

int modificarCertificado(TrCertificado cert) throws TrException

Modifica un certificado existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrCertificado cert Certificado a modificar

Salida

int Número de filas modificadas

2.4.2.2.1.3 Obtener certificados

TrCertificado[] obtenerCertificados(TpoPK idCert,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios certificados. Si se pasa el parámetro *idCert* se devuelve los datos de ese certificado, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idCert Identificador del certificado a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrCertificado

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrCertificado

Salida

TrCertificado[] Array de objetos TrCertificado

2.4.2.2.2 Mantenimiento de metafases

2.4.2.2.2.1 Insertar metafase

TpoPK insertarMetafase(TrMetafase metafase) throws TrException

Inserta una nueva metafase. Si todo es correcto devuelve el identificador de la nueva metafase, si algo falla lanza una excepción.

Entradas

TrMetafase metafase Metafase a insertar

Salida

TpoPK Identificador de la nueva metafase, cero si no se ha insertado

2.4.2.2.2.2 Modificar metafase

int modificarMetafase(TrMetafase metafase) throws TrException

Modifica una metafase existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrMetafase metafase Metafase a modificar

Salida

int Número de filas modificadas

2.4.2.2.2.3 Eliminar metafase

int eliminarMetafase(TpoPK idMetafase) throws TrException

Elimina una metafase existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idMetafase Identificador de la metafase a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.2.4 Obtener metafase

TrMetafase[] obtenerMetafase(TpoPK idMetafase, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias metafases. Si se pasa el parámetro *idMetafase* se devuelve los datos de esa metafase, si se pasa *null*, se devuelven todas.

Entradas

TpoPK idMetafase Identificador de la metafase a obtener. Si se pasa *null* se obtienen todas.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrMetafase

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrMetafase

Salida

TrMetafase[] Array de objetos TrMetafase

int obtenerMetafaseCount(TpoPK idMetafase, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Devuelve el número de metafases

Entradas

<i>TpoPK idMetafase</i>	Identificador de la metafase a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrMetafase
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrMetafase

Salida

<i>int</i>	Número de registros
------------	---------------------

2.4.2.2.3 Mantenimiento de Tipo certificados SCSP

2.4.2.2.3.1 Insertar tipo certificado

TpoPK insertarTipoCertificado(TrTipoCertificado tipoCert) throws TrException

Inserta un nuevo tipo certificado. Si todo es correcto devuelve el identificador del nuevo tipo certificado, si algo falla lanza una excepción.

Entradas

TrTipoCertificado tipoCert Tipo de certificado a insertar

Salida

TpoPK Identificador del tipo certificado insertado, cero si no se ha insertado

2.4.2.2.3.2 Modificar tipo certificado

int modificarTipoCertificado(TrTipoCertificado tipoCert) throws TrException

Modifica un tipo certificado existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoCertificado tipoCert Tipo de certificado a modificar

Salida

int Número de filas modificadas

2.4.2.2.3.3 Eliminar tipo certificado

int eliminarTipoCertificado(TpoPK idTipoCert) throws TrException

Elimina un tipo certificado existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoCert Identificador del tipo de certificado a eliminar



Salida

int Número de filas eliminadas

2.4.2.2.3.4 Obtener tipos de certificados

TrTipoCertificado[] obtenerTiposCertificado(TpoPK idTipoCert,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios tipos de certificado. Si se pasa el parámetro *idTipoCert* se devuelve los datos de ese certificado, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoCert Identificador de el tipo de certificado a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoCertificado

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoCertificado

Salida

TrTipoCertificado[] Array de objetos TrTipoCertificado

2.4.2.2.4 Mantenimiento de definiciones de procedimientos

2.4.2.2.4.1 Insertar definición de procedimiento

TpoPK insertarDefProcedimiento(TrDefProcedimiento defProc) throws TrException

Inserta una nueva definición de procedimiento. Si todo es correcto devuelve el identificador de la nueva definición de procedimiento, si algo falla lanza una excepción.

Entradas

TrDefProcedimiento defProc Definición del procedimiento a insertar

Salida

TpoPK Identificador de la nueva definición de procedimiento, cero si no se ha insertado

2.4.2.2.4.2 Modificar definición de procedimiento

int modificarDefProcedimiento(TrDefProcedimiento defProc) throws TrException

Modifica una definición de procedimiento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrDefProcedimiento defProc Definición de procedimiento a modificar

Salida

int Número de filas modificadas

2.4.2.2.4.3 Eliminar definición de procedimiento

int eliminarDefProcedimiento(TpoPK idDefProc) throws TrException

Elimina una definición de procedimiento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idDefProc Identificador de la definición de procedimiento a eliminar

Salida



int

Número de filas eliminadas

2.4.2.2.4.4 Obtener definición de procedimiento

```
TrDefProcedimiento[ ] obtenerDefProcedimiento( TpoPK idDefProc,  
ClausulaWhere where,  
ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener una o varias definiciones de procedimientos. Si se pasa el parámetro *idDefProc* se devuelve los datos de esa definición de procedimiento, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrDefProcedimiento
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrDefProcedimiento

Salida

<i>TrDefProcedimiento[]</i>	Array de objetos TrDefProcedimiento
------------------------------	-------------------------------------

```
int obtenerDefProcedimientoCount( TpoPK idDefProc,  
ClausulaWhere where,  
ClausulaOrderBy orderBy ) throws TrException
```

Devuelve el número de procedimientos definidos.

Entradas

<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrDefProcedimiento
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrDefProcedimiento

Salida

<i>int</i>	Número de registros
------------	---------------------

2.4.2.2.4.5 Bloquear definición de procedimiento

void bloquearDefProcedimiento(TpoPK idDefProc) throws TrException

Bloquea la definición del procedimiento que se indique en el parámetro *idDefProc*, siempre que no esté bloqueada por otro usuario.

Entradas

<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento
------------------------	--

2.4.2.2.4.6 Desbloquear definición de procedimiento

void desbloquearDefProcedimiento(TpoPK idDefProc) throws TrException

Desbloquea la definición del procedimiento que se indique en el parámetro *idDefProc*, siempre que esté bloqueada por el mismo usuario.

Entradas

<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento
------------------------	--

2.4.2.2.4.7 Procedimiento bloqueado por un usuario

**String procedimientoBloqueadoPor(TpoPK idDefProc,
TpoString nombre) throws TrException;**

Permite obtener el usuario que tiene bloqueado el procedimiento.

Entradas

<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento
------------------------	--

Entrada / Salida

<i>TpoString nombre</i>	Nombre y apellidos del usuario
-------------------------	--------------------------------

Salida

<i>String</i>	Identificador del usuario que tiene bloqueado el procedimiento.
---------------	---

2.4.2.2.4.8 Obtener otros datos

String obtenerOtrosDatosDefProcedimiento(TpoPK idDefProc) throws TrException

Permite obtener otros datos de la definición del procedimiento

Entradas

<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento
------------------------	--

Salida

<i>String</i>	Otros datos del procedimiento (típicamente xml).
---------------	--

2.4.2.2.4.9 Actualizar otros datos

**void actualizarOtrosDatosDefProcedimiento(TpoPK idDefProc,
String contenido) throws TrException**

Permite actualizar otros datos de la definición del procedimiento

Entradas

<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento.
------------------------	---

<i>String contenido</i>	Contenido para otros datos del procedimiento.
-------------------------	---

2.4.2.2.5 Mantenimiento de documentos

2.4.2.2.5.1 Eliminar documento de expediente

**void eliminarDocExpedienteAdmin(TrDocumentoExpediente docExpte,
TrDocumentoBorrado docBorr, TpoPK refSis) throws TrException**

Elimina un documento asociado a un expediente que pertenece al sistema indicado. Si algo falla lanza una excepción.

Entradas

<i>TrDocumentoExpediente</i> <i>docExpte</i>	Documento de expediente a eliminar
<i>TrDocumentoBorrado</i> <i>docBorr</i>	Documento justificante del borrado del documento
<i>TpoPK refSis</i>	Identificador del sistema al que pertenece el expediente del documento

2.4.2.2.6 Mantenimiento de expedientes

2.4.2.2.6.1 Eliminar expediente

<p>int eliminarExpediente(TpoPK idExpte, TrDocumentoBorrado docBorr, TpoPK refSis) throws TrException</p>

Elimina un expediente que pertenece al sistema indicado. Si algo falla lanza una excepción.

Entradas

<i>TpoPK idExpte</i>	Identificador del expediente a eliminar
<i>TrDocumentoBorrado docBorr</i>	Documento justificante de la eliminación del expediente
<i>TpoPK refSis</i>	Identificador del sistema al que pertenece el expediente

Salida

<i>int</i>	Número de documentos eliminados del expediente.
------------	---

2.4.2.2.6.2 Deshacer fase del expediente

int deshacerFaseExpediente(
TpoPK idExpXFase
TpoPK idStma) throws TrException

Permite al usuario deshacer el paso dado en la tramitación de un expediente siempre que dicho paso corresponda a una situación actual del expediente.

Si el sistema establecido tiene conexión al bus se informará del cambio de fases del expediente, además de los interesados en el expediente. Si el expediente no había sido informado previamente se informará con todos sus datos en este momento.

Entradas

<i>TpoPK idExpXFase</i>	<i>Identificador del expediente en fase.</i> Si el identificador es 'null', o no válido, se generará una excepción.
<i>TpoPK idStma</i>	<i>Identificador del sistema al que pertenece el expediente.</i>

Salida

<i>int</i>	Número de documentos eliminados del expediente.
------------	---

2.4.2.2.6.3 Reiniciar expediente

int reiniciarExpediente(
TpoPK idExpte,
TpoPK idStma) throws TrException

Reinicia un expediente , eliminando todas sus tareas y documentos, y llevándolo a la primera fase.

Entradas

<i>TpoPK idExpte</i>	Identificador del expediente a reiniciar
<i>TpoPK idStma</i>	Identificador del sistema al que pertenece el expediente

Salida

<i>int</i>	Número de documentos eliminados del expediente.
------------	---

2.4.2.2.7 Mantenimiento de fases

2.4.2.2.7.1 Insertar fase

TpoPK insertarFase(TrFase fase) throws TrException

Inserta una nueva fase. Si todo es correcto devuelve el identificador de la nueva fase, si algo falla lanza una excepción.

Entradas

<i>TrFase fase</i>	Fase a insertar
--------------------	-----------------

Salida

<i>TpoPK</i>	Identificador de la nueva fase, cero si no se ha insertado
--------------	--

2.4.2.2.7.2 Modificar fase

int modificarFase(TrFase fase) throws TrException

Modifica una fase existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrFase fase</i>	Fase a modificar
--------------------	------------------

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.7.3 Eliminar fase

int eliminarFase(TpoPK idFase) throws TrException

Elimina una fase existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idFase</i>	Identificador de la fase a eliminar
---------------------	-------------------------------------

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.7.4 Obtener fase

```
TrFase [ ] obtenerFase( TpoPK idFase,  
                        ClausulaWhere where,  
                        ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener una o varias fases. Si se pasa el parámetro *idFase* se devuelve los datos de esa fase, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idFase</i>	Identificador de la fase a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrFase
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrFase

Salida

<i>TrFase[]</i>	Array de objetos TrFase
------------------	-------------------------

```
int obtenerFaseCount( TpoPK idFase,  
                     ClausulaWhere where,  
                     ClausulaOrderBy orderBy ) throws TrException
```

Devuelve el número de fases

Entradas

<i>TpoPK idFase</i>	Identificador de la fase a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrFase
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrFase

Salida

<i>int</i>	Número de registros
------------	---------------------



2.4.2.2.8 Mantenimiento de transiciones

2.4.2.2.8.1 Insertar transición

```
void insertarTransicionDefProcedimiento(  
    TrTransicionDefProcedimiento transicionDefProc,  
    TpoPK idTransicion,  
    TpoPK idDefProc) throws TrException
```

Inserta una nueva transición asociada a un procedimiento. Si la transición ya existía, simplemente se asocia al procedimiento. Si todo es correcto devuelve el identificador de la nueva transición asociada al procedimiento, si algo falla lanza una excepción.

Entradas

TrTransicionDefProcedimiento transicionDefProc Transición asociada al procedimiento a insertar

Entrada / Salida

TpoPK idTransición Identificador de la transición asociada al procedimiento, cero si no se ha insertado.

TpoPK idDefProc Identificador del procedimiento al que se ha asociado la transición , cero si no se ha insertado.

2.4.2.2.8.2 Modificar transición

```
int modificarTransicionDefProcedimiento(  
    TrTransicionDefProcedimiento transicionDefProc ) throws TrException
```

Modifica una transición asociada a un procedimiento existente. Si se modifican los datos de la transición estos afectan a todos los procedimientos. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTransicionDefProcedimiento transicionDefProc transición asociada a un procedimiento a modificar

Salida

int Número de filas modificadas

2.4.2.2.8.3 Eliminar transición

int eliminarTransicionDefProcedimiento(TpoPK idTransicion, TpoPK idDefProc) throws TrException

Elimina una transición asociada a un procedimiento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idTransicion</i>	Identificador de la transición
<i>TpoPK idDefProc</i>	Identificador del procedimiento

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.8.4 Obtener transición

TrTransicionDefProcedimiento[] obtenerTransicionDefProcedimiento(TpoPK idTransicion, TpoPK idDefProc, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias transiciones. Si se pasa el parámetro *idTransición* se devuelve los datos de esa transición asociada al procedimiento que se indique en el parámetro *idDefProc*, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idTransición</i>	Identificador de la transición. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idDefProc</i>	Identificador del procedimiento. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrTransicionDefProcedimiento
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrTransicionDefProcedimiento

Salida

<i>TrTransicionDefProcedimiento[]</i>	Array de objetos TrTransicionDefProcedimiento
--	---

2.4.2.2.8.5 Ajustar transición

```
int ajustarTransicionExpedientesFase(
    TpoPK idTransicion,
    TpoPK idTransicionAjuste) throws TrException
```

Permite ajustar la transición de los expedientes en fase. Este método es útil cuando se invalida una transición y se crea una nueva y queremos que los expedientes en fase ya existentes usen la nueva transición.

Entradas

<i>TpoPK</i> idTransicion	Identificador de la transición. Si se pasa <i>null</i> o una transición inválida generará una excepción.
<i>TpoPK</i> idTransicionAjuste	Identificador de la nueva transición. Si se pasa <i>null</i> o una transición inválida generará una excepción.

Salida

<i>int</i>	Número de filas ajustadas
------------	---------------------------

2.4.2.2.9 Mantenimiento de perfiles de usuario

2.4.2.2.9.1 Insertar perfil de usuario

TpoPK insertarPerfilUsuario(TrPerfilUsuario perfilUsuario) throws TrException

Inserta un nuevo perfil de usuario. Si todo es correcto devuelve el identificador del nuevo perfil de usuario, si algo falla lanza una excepción.

Entradas

TrPerfilUsuario perfilUsuario Perfil de usuario a insertar

Salida

TpoPK Identificador del nuevo perfil de usuario, cero si no se ha insertado

2.4.2.2.9.2 Modificar perfil de usuario

int modificarPerfilUsuario(TrPerfilUsuario perfilUsuario) throws TrException

Modifica un perfil de usuario existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrPerfilUsuario perfilUsuario Perfil de usuario a modificar

Salida

int Número de filas modificadas

2.4.2.2.9.3 Eliminar perfil de usuario

int eliminarPerfilUsuario(TpoPK idPerfilUsu) throws TrException

Elimina un perfil de usuario existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idPerfilUsu Identificador del perfil de usuario a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.9.4 Obtener perfil de usuario

```
TrPerfilUsuario [ ] obtenerPerfilUsuario( TpoPK idPerfilUsu,  
 ClausulaWhere where,  
 ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener uno o varios perfiles de usuario. Si se pasa el parámetro *idPerfilUsu* se devuelve los datos de ese perfil de usuario, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idPerfilUsu</i>	Identificador del perfil de usuario a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrPerfilUsuario
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrPerfilUsuario

Salida

<i>TrPerfilUsuario[]</i>	Array de objetos TrPerfilUsuario
---------------------------	----------------------------------

2.4.2.2.10 Mantenimiento de tipos de expediente

2.4.2.2.10.1 Insertar tipo de expediente

```
TpoPK insertarTipoExpediente( TrTipoExpediente tipoExp ) throws TrException
```

Inserta un nuevo tipo de expediente. Si todo es correcto devuelve el identificador del nuevo tipo de expediente, si algo falla lanza una excepción.

Entradas

<i>TrTipoExpediente tipoExp</i>	Tipo de expediente a insertar
---------------------------------	-------------------------------

Salida

<i>TpoPK</i>	Identificador del nuevo tipo de expediente, cero si no se ha insertado
--------------	--

2.4.2.2.10.2 Modificar tipo de expediente

int modificarTipoExpediente(TrTipoExpediente tipoExp) throws TrException

Modifica un tipo de expediente existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoExpediente tipoExp Tipo de expediente a modificar

Salida

int Número de filas modificadas

2.4.2.2.10.3 Eliminar tipo de expediente

int eliminarTipoExpediente(TpoPK idTipoExp) throws TrException

Elimina un tipo de expediente existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoExp Identificador del tipo de expediente a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.10.4 Obtener tipo de expediente

**TrTipoExpediente [] obtenerTipoExpediente(TpoPK idTipoExp,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException**

Permite obtener uno o varios tipos de expediente. Si se pasa el parámetro *idTipoExp* se devuelve los datos de ese tipo de expediente, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoExp Identificador del tipo de expediente a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoExpediente

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoExpediente

Salida

TrTipoExpediente[] Array de objetos TrTipoExpediente

int obtenerTipoExpedienteCount(TpoPK idTipoExp,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve el número de tipos de expedientes

Entradas

TpoPK idTipoExp Identificador del tipo de expediente a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoExpediente

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoExpediente

Salida

int Número de registros

2.4.2.2.11 Mantenimiento de versiones de procedimientos

2.4.2.2.11.1 Insertar versión de procedimiento

```
void insertarVersionDefProcedimiento(
    TrVersionDefProcedimiento versionDefProc,
    TpoPK idDefProc,
    TpoPK idTipoExp) throws TrException
```

Inserta una nueva versión de un procedimiento asociado a un tipo de expediente. Si todo es correcto devuelve el identificador del procedimiento y del tipo de expediente, si algo falla lanza una excepción.

Entradas

<i>TrVersionDefProcedimiento</i> versionDefProc	Versión del procedimiento a insertar
--	--------------------------------------

Entrada / Salida

<i>TpoPK</i> idDefProc	Identificador del procedimiento, cero si no se ha insertado la versión.
<i>TpoPK</i> idTipoExp	Identificador del tipo de expediente, cero si no se ha insertado la versión.

2.4.2.2.11.2 Modificar versión de procedimiento

```
int modificarVersionDefProcedimiento(
    TrVersionDefProcedimiento versionDefProc ) throws TrException
```

Modifica una versión de procedimiento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrVersionDefProcedimiento</i> versionDefProc	Versión del procedimiento a modificar
--	---------------------------------------

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.11.3 Eliminar versión de procedimiento

```
int eliminarVersionDefProcedimiento( TpoPK idDefProc,
```

TpoPK idTipoExp) throws TrException

Elimina una versión de procedimiento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idDefProc</i>	Identificador del procedimiento
<i>TpoPK idTipoExp</i>	Identificador del tipo de expediente

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.11.4 Obtener versión de procedimiento

TrVersionDefProcedimiento[] obtenerVersionDefProcedimiento(
TpoPK idDefProc,
TpoPK idTipoExp,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias versiones de procedimiento. Si se pasa el parámetro idDefProc se devuelven las versiones de ese procedimiento y si se para el parámetro *idTipoExp* se devuelven las asociadas a ese tipo de expediente, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idDefProc</i>	Identificador del procedimiento. Si se pasa <i>null</i> se obtienen todas las versiones asociadas a ese procedimiento.
<i>TpoPK idTipoExp</i>	Identificador del tipo de expediente. Si se pasa <i>null</i> se obtienen todas las versiones asociadas a ese tipo de expediente.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrVersionDefProcedimiento
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrVersionDefProcedimiento

Salida

TrVersionDefProcedimiento []	Array de objetos TrVersionDefProcedimiento
-------------------------------	--

2.4.2.2.12 Mantenimiento de extremos de transiciones gráficas

2.4.2.2.12.1 Insertar extremo de transición gráfica

TpoPK insertarExtremoTransicionGr(TrExtremoTransicionGr extremoTransGr) throws
TrException

Inserta un nuevo extremo de una transición gráfica. Si todo es correcto devuelve el identificador del nuevo extremo de la transición gráfica, si algo falla lanza una excepción.

Entradas

TrExtremoTransicionGr extremoTransGr	Extremo de una transición gráfica a insertar
---	--

Salida

TpoPK	Identificador del nuevo extremo de una transición gráfica, cero si no se ha insertado
-------	---

2.4.2.2.12.2 Modificar extremo de transición gráfica

int modificarExtremoTransicionGr(TrExtremoTransicionGr extremoTransGr) throws
TrException

Modifica un extremo de una transición gráfica existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrExtremoTransicionGr extremoTransGr	Extremo de una transición gráfica a modificar
---	---

Salida

int	Número de filas modificadas
-----	-----------------------------

2.4.2.2.12.3 Eliminar extremo de transición gráfica

int eliminarExtremoTransicionGr(TpoPK idExtremoTransGr) throws TrException

Elimina un extremo de una transición gráfica existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idExtremoTransGr	Identificador del extremo de una transición gráfica a eliminar
------------------------	--

Salida

int	Número de filas eliminadas
-----	----------------------------

2.4.2.2.12.4 Obtener extremo de transición gráfica

```

TrExtremoTransicionGr [ ] obtenerExtremoTransicionGr(
    TpoPK idExtremoTransGr,
    ClausulaWhere where,
    ClausulaOrderBy orderBy ) throws TrException

```

Permite obtener uno o varios extremos de una o varias transiciones gráficas. Si se pasa el parámetro *idExtremoTransGr* se devuelve los datos de ese extremo de la transición gráfica, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK idExtremoTransGr</i>	Identificador del extremo de la transición gráfica a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrExtremoTransicionGr
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrExtremoTransicionGr

Salida

<i>TrExtremoTransicionGr[]</i>	Array de objetos TrExtremoTransicionGr
---------------------------------	--

2.4.2.2.13 Mantenimiento de definición de procedimientos gráficos

2.4.2.2.13.1 Insertar definición de procedimiento gráfico

TpoPK insertarDefProcedimientoGr(TrDefProcedimientoGr defProcGr) throws TrException

Inserta una nueva definición de procedimiento gráfico. Si todo es correcto devuelve el identificador de la nueva definición de procedimiento gráfico, si algo falla lanza una excepción.

Entradas

<i>TrDefProcedimientoGr</i> <i>defProcGr</i>	Definición de procedimiento gráfico a insertar
---	--

Salida

<i>TpoPK</i>	Identificador de la nueva definición de procedimiento gráfico, cero si no se ha insertado
--------------	---

2.4.2.2.13.2

2.4.2.2.13.3 Modificar definición de procedimiento gráfico

int modificarDefProcedimientoGr(TrDefProcedimientoGr defProcGr) throws TrException

Modifica una definición de procedimiento gráfico existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrDefProcedimientoGr</i> <i>defProcGr</i>	Definición de procedimiento gráfico a modificar
---	---

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.13.4 Eliminar definición de procedimiento gráfico

int eliminarDefProcedimientoGr(TpoPK idDefProcGr) throws TrException

Elimina una definición de procedimiento gráfico existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idDefProcGr</i>	Identificador de la definición de procedimiento gráfico a eliminar
--------------------------	--

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.13.5 Obtener definición de procedimiento gráfico

TrDefProcedimientoGr [] obtenerDefProcedimientoGr(TpoPK idDefProcGr,

ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias definiciones de procedimientos gráficos. Si se pasa el parámetro *idDefProcGr* se devuelve los datos de esa definición de procedimiento gráfico, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK</i> idDefProcGr	Identificador de la definición de procedimiento gráfico a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrDefProcedimientoGr
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrDefProcedimientoGr

Salida

<i>TrDefProcedimientoGr</i> []	Array de objetos TrDefProcedimientoGr
--------------------------------	---------------------------------------

2.4.2.2.14 Mantenimiento de metafases gráficas

2.4.2.2.14.1 Insertar metafase gráfica

TpoPK insertarMetafaseGr(TrMetafaseGr metafaseGr) throws TrException

Inserta una nueva metafase gráfica. Si todo es correcto devuelve el identificador de la nueva metafase gráfica, si algo falla lanza una excepción.

Entradas

TrMetafaseGr metafaseGr Metafase gráfica a insertar

Salida

TpoPK Identificador de la nueva metafase gráfica, cero si no se ha insertado

2.4.2.2.14.2 Modificar metafase gráfica

int modificarMetafaseGr(TrMetafaseGr metafaseGr) throws TrException

Modifica una metafase gráfica existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrMetafaseGr metafaseGr Metafase gráfica a modificar

Salida

int Número de filas modificadas

2.4.2.2.14.3 Eliminar metafase gráfica

int eliminarMetafaseGr(TpoPK idMetafaseGr) throws TrException

Elimina una metafase gráfica existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idMetafaseGr Identificador de la metafase gráfica a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.14.4 Obtener metafase gráfica

TrMetafaseGr [] obtenerMetafaseGr(TpoPK idMetafaseGr, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias metafases gráficas. Si se pasa el parámetro *idMetafaseGr* se devuelve los datos de esa metafase gráfica, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idMetafaseGr</i>	Identificador de la metafase gráfica a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrMetafaseGr
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrMetafaseGr

Salida

<i>TrMetafaseGr[]</i>	Array de objetos TrMetafaseGr
------------------------	-------------------------------

2.4.2.2.15 Mantenimiento de nodos de transición gráfica

2.4.2.2.15.1 Insertar nodo de transición gráfica

TpoPK insertarNodoTransicionGr(TrNodoTransicionGr nodoTransGr) throws TrException

Inserta un nuevo nodo de transición gráfica. Si todo es correcto devuelve el identificador del nuevo nodo de transición gráfica, si algo falla lanza una excepción.

Entradas

<i>TrNodoTransicionGr nodoTransGr</i>	Nodo de transición gráfica a insertar
---	---------------------------------------

Salida

<i>TpoPK</i>	Identificador del nuevo nodo de transición gráfica, cero si no se ha insertado
--------------	--

2.4.2.2.15.2 Modificar nodo de transición gráfica



int modificarNodoTransicionGr(TrNodoTransicionGr nodoTransGr) throws TrException

Modifica un nodo de transición gráfica existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrNodoTransicionGr Nodo de transición gráfica a modificar
nodoTransGr

Salida

int Número de filas modificadas

2.4.2.2.15.3 Eliminar nodo de transición gráfica

int eliminarNodoTransicionGr(TpoPK idNodoTransGr) throws TrException

Elimina un nodo de transición gráfica existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idNodoTransGr Identificador del nodo de transición gráfica a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.15.4 Obtener nodo de transición gráfica

TrNodoTransicionGr [] obtenerNodoTransicionGr(TpoPK idNodoTransGr, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios nodos de las transiciones gráficas. Si se pasa el parámetro *idNodoTransGr* se devuelve los datos de ese nodo de transición gráfica, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idNodoTransGr Identificador del nodo de transición gráfica a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para *TrNodoTransicionGr*

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para *TrNodoTransicionGr*

Salida

TrNodoTransicionGr [] Array de objetos *TrNodoTransicionGr*

2.4.2.2.16 Mantenimiento de transiciones gráficas

2.4.2.2.16.1 Insertar transición gráfica

TpoPK insertarTransicionGr(TrTransicionGr transicionGr) throws TrException

Inserta una nueva transición gráfica. Si todo es correcto devuelve el identificador de la nueva transición gráfica, si algo falla lanza una excepción.

Entradas

TrTransicionGr transicionGr Transición gráfica a insertar

Salida

TpoPK Identificador de la nueva transición gráfica, cero si no se ha insertado

2.4.2.2.16.2 Modificar transición gráfica

int modificarTransicionGr(TrTransicionGr transicionGr) throws TrException

Modifica una transición gráfica existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTransicionGr transicionGr transición gráfica a modificar

Salida

int Número de filas modificadas

2.4.2.2.16.3 Eliminar transición gráfica

int eliminarTransicionGr(TpoPK idTransicionGr) throws TrException

Elimina una transición gráfica existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTransicionGr Identificador de la transición gráfica a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.16.4 Obtener transición gráfica

TrTransicionGr [] obtenerTransicionGr(TpoPK idTransicionGr, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias transiciones gráficas. Si se pasa el parámetro *idTransicionGr* se devuelve los datos de esa transición gráfica, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idTransicionGr</i>	Identificador de la transición gráfica a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrTransicionGr
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrTransicionGr

Salida

<i>TrTransicionGr[]</i>	Array de objetos TrTransicionGr
--------------------------	---------------------------------

2.4.2.2.17 Mantenimiento de tareas (Manipulación datos y otras)

2.4.2.2.17.1 Insertar tarea

TpoPK insertarBloque(TrBloque bloque) throws TrException

Inserta una nueva tarea o bloque. Si todo es correcto devuelve el identificador de la nueva tarea o bloque, si algo falla lanza una excepción.

Entradas

TrBloque bloque Tarea o bloque a insertar

Salida

TpoPK Identificador de la nueva tarea o bloque, cero si no se ha insertado

2.4.2.2.17.2 Modificar tarea

int modificarBloque(TrBloque bloque) throws TrException

Modifica una tarea o bloque existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrBloque bloque Tarea o bloque a modificar

Salida

int Número de filas modificadas

2.4.2.2.17.3 Eliminar tarea

int eliminarBloque(TpoPK idBloque) throws TrException

Elimina una tarea o bloque existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idBloque Identificador de la tarea o bloque a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.17.4 Obtener tarea

TrBloque [] obtenerBloque(TpoPK idBloque, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias tareas o bloques. Si se pasa el parámetro *idBloque* se devuelve los datos de esa tarea o bloque, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idBloque</i>	Identificador de la tarea o bloque a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrBloque
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrBloque

Salida

<i>TrBloque[]</i>	Array de objetos TrBloque
-------------------	---------------------------

2.4.2.2.18 Mantenimiento de tareas en fase (Manipulación datos y otras)

2.4.2.2.18.1 Insertar tarea en fase

void insertarBloquePermitidoDefProc(TrBloquePermitidoDefProc bloquePer, TpoPK idBloquePer, TpoPK idDefProc) throws TrException

Inserta una nueva tarea en fase o bloque permitido. Si el bloque permitido ya existía se asocia al procedimiento. Si todo es correcto devuelve el identificador de la nueva tarea en fase o bloque permitido, si algo falla lanza una excepción.

Entradas

TrBloquePermitidoDefProc bloquePer	tarea en fase o bloque permitido a insertar
---------------------------------------	---

Entrada / Salida

<i>TpoPK idBloquePer</i>	Identificador de la nueva tarea en fase o bloque permitido, cero si no se ha insertado
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento, cero si no se ha insertado

2.4.2.2.18.2 Modificar tarea en fase

int modificarBloquePermitidoDefProc(TrBloquePermitidoDefProc bloquePer) throws TrException

Modifica una tarea en fase o bloque permitido existente. Si el bloque permitido está asociada a varios procedimientos los cambios afectan a todos ellos. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrBloquePermitidoDefProc</i> <i>bloquePer</i>	Tarea en fase o bloque permitido a modificar
---	--

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.18.3 Eliminar tarea en fase

int eliminarBloquePermitidoDefProc (TpoPK idBloquePer, TpoPK idDefProc) throws TrException

Elimina una tarea en fase o bloque permitido existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idBloquePer</i>	Identificador de la tarea en fase o bloque permitido a eliminar
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento.

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.18.4 Obtener tarea en fase

TrBloquePermitidoDefProc [] obtenerBloquePermitidoDefProc(
 TpoPK idBloquePer,
 TpoPK idDefProc,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias tareas en fase o bloques permitidos. Si se pasa el parámetro idBloquePer se devuelve los datos de esa tarea en fase o bloque permitido, y si se pasa el parámetro idDefProc se devuelven los que estén asociados a ese procedimiento, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idBloquePer</i>	Identificador de la tarea en fase o bloque permitido a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrBloquePermitidoDefProc
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrBloquePermitidoDefProc

Salida

TrBloquePermitidoDefProc[] Array de objetos TrBloquePermitidoDefProc

2.4.2.2.19 Mantenimiento de caducidades

2.4.2.2.19.1 Insertar caducidad

TpoPK insertarCaducidad(TrCaducidad caducidad) throws TrException

Inserta una nueva caducidad. Si todo es correcto devuelve el identificador de la nueva caducidad, si algo falla lanza una excepción.

Entradas

TrCaducidad caducidad	Caducidad a insertar
-----------------------	----------------------

Salida

<i>TpoPK</i>	Identificador de la nueva caducidad, cero si no se ha insertado
--------------	---

2.4.2.2.19.2 Modificar caducidad

int modificarCaducidad(TrCaducidad caducidad) throws TrException

Modifica una caducidad existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrCaducidad caducidad Caducidad a modificar

Salida

int Número de filas modificadas

2.4.2.2.19.3 Eliminar caducidad

int eliminarCaducidad(TpoPK idCaducidad) throws TrException

Elimina una caducidad existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idCaducidad Identificador de la caducidad a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.19.4 Obtener caducidad

TrCaducidad [] obtenerCaducidad(TpoPK idCaducidad, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias caducidades. Si se pasa el parámetro *idCaducidad* se devuelve los datos de esa caducidad, si se pasa *null*, se devuelven todas.

Entradas

TpoPK idCaducidad Identificador de la caducidad a obtener. Si se pasa *null* se obtienen todas.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrCaducidad

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrCaducidad

Salida

TrCaducidad[] Array de objetos TrCaducidad

2.4.2.2.20 Mantenimiento de tareas (Manipulación de escritos)

2.4.2.2.20.1 Insertar tipo de documento

TpoPK insertarTipoDocumento(TrTipoDocumento tipoDoc) throws TrException

Inserta una nueva tarea o tipo de documento. Si todo es correcto devuelve el identificador de la nueva tarea o tipo de documento, si algo falla lanza una excepción.

Entradas

TrTipoDocumento tipoDoc Tarea o tipo de documento a insertar

Salida

TpoPK Identificador de la nueva tarea o tipo de documento, cero si no se ha insertado

2.4.2.2.20.2 Modificar tipo de documento

int modificarTipoDocumento(TrTipoDocumento tipoDoc) throws TrException

Modifica una tarea o tipo de documento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoDocumento tipoDoc Tarea o tipo de documento a modificar

Salida

int Número de filas modificadas

2.4.2.2.20.3 Eliminar tipo de documento

int eliminarTipoDocumento(TpoPK idTipoDoc) throws TrException

Elimina una tarea o tipo de documento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoDoc Identificador de la tarea o tipo de documento a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.20.4 Obtener tipo de documento

```
TrTipoDocumento [ ] obtenerTipoDocumento( TpoPK idTipoDoc,
                                           ClausulaWhere where,
                                           ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener una o varias tareas o tipos de documentos Si se pasa el parámetro *idTipoDoc* se devuelve los datos de esa tarea o tipo de documento, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idTipoDoc</i>	Identificador de la tarea o tipo de documento a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrTipoDocumento
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrTipoDocumento

Salida

<i>TrTipoDocumento[]</i>	Array de objetos TrTipoDocumento
---------------------------	----------------------------------

2.4.2.2.21 Mantenimiento de tareas en fase (Manipulación de escritos)

2.4.2.2.21.1 Insertar tarea en fase

```
void insertarDocumentoPermitidoDefProc(
    TrDocumentoPermitidoDefProc docPermitido,
    TpoPK idDocumentoPer,
    TpoPK idDefProc ) throws TrException
```

Inserta una nueva tarea en fase o documento permitido. Si el documento permitido ya existía se asocia al procedimiento. Si todo es correcto devuelve el identificador de la nueva tarea en fase o documento permitido, si algo falla lanza una excepción.

Entradas

TrDocumentoPermitidoDefProc docPermitido Tarea en fase o documento permitido a insertar

Entrada / Salida

TpoPK idDocumentoPer Identificador de la nueva tarea en fase o documento permitido, cero si no se ha insertado

TpoPK idDefProc Identificador de la definición del procedimiento, cero si no se ha insertado

2.4.2.2.21.2 Modificar tarea en fase

```
int modificarDocumentoPermitidoDefProc(
    TrDocumentoPermitidoDefProc docPermitido ) throws TrException
```

Modifica una tarea en fase o documento permitido existente. Si el documento permitido está asociada a varios procedimientos los cambios afectan a todos ellos. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrDocumentoPermitidoDefProc docPermitido Tarea en fase o documento permitido a modificar

Salida

int Número de filas modificadas

2.4.2.2.21.3 Eliminar tarea en fase

```
int eliminarDocumentoPermitidoDefProc ( TpoPK idDocumentoPer,
```

TpoPK idDefProc) throws TrException

Elimina una tarea en fase o documento permitido existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idDocumentoPer</i>	Identificador de la tarea en fase o documento permitido a eliminar
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento.

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.1.4 Obtener tarea en fase

TrDocumentoPermitidoDefProc [] obtenerDocumentoPermitidoDefProc(
TpoPK idDocumentoPer,
TpoPK idDefProc,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias tareas en fase o documentos permitidos. Si se pasa el parámetro *idDocumentoPer* se devuelve los datos de esa tarea en fase o documento permitido, y si se pasa el parámetro *idDefProc* se devuelven los que estén asociados a ese procedimiento, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idDocumentoPer</i>	Identificador de la tarea en fase o documento permitido a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrDocumentoPermitidoDefProc
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrDocumentoPermitidoDefProc

Salida

<i>TrDocumentoPermitidoDefProc[]</i>	Array de objetos TrDocumentoPermitidoDefProc
---------------------------------------	--

2.4.2.2.22 Mantenimiento de condiciones

2.4.2.2.22.1 Insertar condición

TpoPK insertarCondicion(TrCondicion condicion) throws TrException

Inserta una nueva condición. Si todo es correcto devuelve el identificador de la nueva condición, si algo falla lanza una excepción.

Entradas

TrCondicion condicion	Condición a insertar
-----------------------	----------------------

Salida

TpoPK	Identificador de la nueva condición, cero si no se ha insertado
-------	---

2.4.2.2.22.2 Modificar condición

int modificarCondicion(TrCondicion condicion) throws TrException

Modifica una condición existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrCondicion condicion	Condición a modificar
-----------------------	-----------------------

Salida

int	Número de filas modificadas
-----	-----------------------------

2.4.2.2.22.3 Eliminar condición

int eliminarCondicion(TpoPK idCondicion) throws TrException

Elimina una condición existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idCondicion	Identificador de la condición a eliminar
-------------------	--

Salida

int	Número de filas eliminadas
-----	----------------------------

2.4.2.2.22.4 Obtener condición

TrCondicion [] obtenerCondicion(TpoPK idCondicion, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias condiciones. Si se pasa el parámetro *idCondicion* se devuelve los datos de esa condición, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idCondicion</i>	Identificador de la condición a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrCondicion
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrCondicion

Salida

<i>TrCondicion[]</i>	Array de objetos TrCondicion
-----------------------	------------------------------

2.4.2.2.23 Mantenimiento de acciones

2.4.2.2.23.1 Insertar acción

TpoPK insertarAccion(TrAccion accion) throws TrException

Inserta una nueva acción. Si todo es correcto devuelve el identificador de la nueva Acción, si algo falla lanza una excepción.

Entradas

<i>TrAccion accion</i>	Acción a insertar
------------------------	-------------------

Salida

<i>TpoPK</i>	Identificador de la nueva acción, cero si no se ha insertado
--------------	--

2.4.2.2.23.2 Modificar acción

int modificarAccion(TrAccion accion) throws TrException

Modifica una acción existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrAccion accion</i>	Acción a modificar
------------------------	--------------------

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.23.3 Eliminar acción

int eliminarAccion(TpoPK idAccion) throws TrException

Elimina una acción existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idAccion</i>	Identificador de la acción a eliminar
-----------------------	---------------------------------------

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.23.4 Obtener acción

TrAccion [] obtenerAccion(TpoPK idAccion,

ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias acciones. Si se pasa el parámetro *idAccion* se devuelve los datos de esa acción, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idAccion</i>	Identificador de la acción a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrAccion
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrAccion

Salida

<i>TrAccion[]</i>	Array de objetos TrAccion
--------------------	---------------------------

2.4.2.2.24 Mantenimiento de avisos

2.4.2.2.24.1 Insertar aviso



TpoPK insertarAviso(TrAviso aviso) throws TrException

Inserta un nuevo aviso. Si todo es correcto devuelve el identificador de la nueva aviso, si algo falla lanza una excepción.

Entradas

<i>TrAviso aviso</i>	Aviso a insertar
----------------------	------------------

Salida

<i>TpoPK</i>	Identificador de la nueva aviso, cero si no se ha insertado
--------------	---

2.4.2.2.24.2 Modificar aviso

int modificarAviso(TrAviso aviso) throws TrException

Modifica un aviso existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrAviso aviso</i>	Aviso a modificar
----------------------	-------------------

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.24.3 Eliminar aviso

int eliminarAviso(TpoPK idAviso) throws TrException

Elimina un aviso existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idAviso</i>	Identificador del aviso a eliminar
----------------------	------------------------------------

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.24.4 Obtener aviso

TrAviso [] obtenerAviso(TpoPK idAviso,

ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios avisos. Si se pasa el parámetro *idAviso* se devuelve los datos de ese aviso, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK idAviso</i>	Identificador del aviso a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrAviso
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrAviso

Salida

<i>TrAviso[]</i>	Array de objetos TrAviso
-------------------	--------------------------

2.4.2.2.25 Mantenimiento de condiciones de una transición

2.4.2.2.25.1 Insertar condición de una transición

void insertarCondicionTransicion(TrCondicionTransicion condicionTrans, TpoPK idCondicion, TpoPK idTransicion, TpoPK idDefProc) throws TrException

Permite asociar una condición a una transición. Si todo es correcto devuelve el identificador de la condición asociada a la transición, si algo falla lanza una excepción.

Entradas

<i>TrCondicionTransicion</i> condicionTrans	Condición asociada a una transición a insertar
--	--

Entrada / Salida

<i>TpoPK</i> idCondicion	Identificador de la condición asociada a la transición, cero si no se ha insertado
<i>TpoPK</i> idTransicion	Identificador de la transición, cero si no se ha insertado
<i>TpoPK</i> idDefProc	Identificador de la definición de procedimiento, cero si no se ha insertado

2.4.2.2.25.2 Modificar condición de una transición

int modificarCondicionTransicion(TrCondicionTransicion condicionTrans) throws TrException

Modifica una condición asociada a una transición existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrCondicionTransicion</i> condicionTrans	Condición asociada a una transición a modificar
--	---

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.25.3 Eliminar condición de una transición

int eliminarCondicionTransicion(TpoPK idCondicion, TpoPK idTransicion, TpoPK idDefProc) throws TrException

Elimina la asociación entre una condición y una transición existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idCondicion</i>	Identificador de la condición asociada a una transición a eliminar
<i>TpoPK idTransicion</i>	Identificador de la transición
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.25.4 Obtener condición de una transición

<p>TrCondicionTransicion[] obtenerCondicionTransicion(TpoPK idCondicion, TpoPK idTransicion, TpoPK idDefProc, ClausulaWhere where,</p>

ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias condiciones asociadas a una transición. Si se pasa el parámetro *idCondicion* se devuelve los datos de esa condición asociada a la transición, si se pasa el parámetro *idTransicion* se filtra por la transición indicada, igualmente pasa con el parámetro *idDefProc* para la definición del procedimiento, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idCondicion</i>	Identificador de la condición. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idTransicion</i>	Identificador de la transición. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrCondicionTransicion
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrCondicionTransicion

Salida

<i>TrCondicionTransicion[]</i>	Array de objetos TrCondicionTransicion
--------------------------------	--

2.4.2.2.26 Mantenimiento de acciones de una transición

2.4.2.2.26.1 Insertar acción de una transición

<pre>void insertarAccionTransicion(TrAccionTransicion accionTrans, TpoPK idAccion, TpoPK idTransicion,</pre>



TpoPK idDefProc) throws TrException

Permite asociar una acción a una transición. Si todo es correcto devuelve el identificador de la acción asociada a la transición, si algo falla lanza una excepción.

Entradas

TrAccionTransicion
accionTrans

Acción asociada a una transición a insertar

Entrada / Salida

TpoPK idAccion

Identificador de la acción asociada a la transición, cero si no se ha insertado

TpoPK idTransicion

Identificador de la transición, cero si no se ha insertado

TpoPK idDefProc

Identificador de la definición de procedimiento, cero si no se ha insertado

2.4.2.2.26.2 Modificar acción de una transición

int modificarAccionTransicion(

TrAccionTransicion accionTrans) throws TrException

Modifica una acción asociada a una transición existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrAccionTransicion
accionTrans

Acción asociada a una transición a modificar

Salida

int

Número de filas modificadas

2.4.2.2.26.3 Eliminar acción de una transición

int eliminarAccionTransicion(*TpoPK idAccion,*

TpoPK idTransicion,

TpoPK idDefProc) throws TrException

Elimina la asociación entre una acción y una transición existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idAccion</i>	Identificador de la acción asociada a una transición a eliminar
<i>TpoPK idTransicion</i>	Identificador de la transición
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.26.4 Obtener acción de una transición

<p>TrAccionTransicion[] obtenerAccionTransicion(TpoPK idAccion, TpoPK idTransicion, TpoPK idDefProc, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Permite obtener una o varias acciones asociadas a una transición. Si se pasa el parámetro *idAccion* se devuelve los datos de esa acción asociada a la transición, si se pasa el parámetro *idTransicion* se filtra por la transición indicada, igualmente pasa con el parámetro *idDefProc* para la definición del procedimiento, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idAccion</i>	Identificador de la acción. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idTransicion</i>	Identificador de la transición. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrAccionTransicion
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrAccionTransicion

Salida

<i>TrAccionTransicion[]</i>	Array de objetos TrAccionTransicion
------------------------------	-------------------------------------

2.4.2.2.27 Mantenimiento de avisos de una transición

2.4.2.2.27.1 Insertar aviso de una transición

```
void insertarAvisoTransicion( TrAvisoTransicion avisoTrans,
                             TpoPK idAviso,
                             TpoPK idTransicion,
                             TpoPK idDefProc) throws TrException
```

Permite asociar un aviso a una transición. Si todo es correcto devuelve el identificador del aviso asociada a la transición, si algo falla lanza una excepción.

Entradas

TrAvisoTransicion avisoTrans Aviso asociada a una transición a insertar

Entrada / Salida

TpoPK idAviso Identificador del aviso asociada a la transición, cero si no se ha insertado

TpoPK idTransicion Identificador de la transición, cero si no se ha insertado

TpoPK idDefProc Identificador de la definición de procedimiento, cero si no se ha insertado

2.4.2.2.27.2 Modificar aviso de una transición

<p>int modificarAvisoTransicion(<i>TrAvisoTransicion</i> avisoTrans) throws TrException</p>

Modifica un aviso asociado a una transición existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrAvisoTransicion avisoTrans Aviso asociada a una transición a modificar

Salida

int Número de filas modificadas

2.4.2.2.27.3 Eliminar aviso de una transición

<p>int eliminarAvisoTransicion(<i>TpoPK</i> idAviso, <i>TpoPK</i> idTransicion, <i>TpoPK</i> idDefProc) throws TrException</p>
--

Elimina la asociación entre un aviso y una transición existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idAviso Identificador del aviso asociada a una transición a eliminar

TpoPK idTransicion Identificador de la transición

TpoPK idDefProc Identificador de la definición de procedimiento

Salida

int Número de filas eliminadas

2.4.2.2.27.4 Obtener aviso de una transición

<p>TrAvisoTransicion[] obtenerAvisoTransicion(TpoPK idAviso, TpoPK idTransicion, TpoPK idDefProc, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>

Permite obtener uno o varios avisos asociadas a una transición. Si se pasa el parámetro *idAviso* se devuelve los datos de ese aviso asociada a la transición, si se pasa el parámetro *idTransicion* se filtra

por la transición indicada, igualmente pasa con el parámetro *idDefProc* para la definición del procedimiento, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idAviso</i>	Identificador del aviso. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idTransicion</i>	Identificador de la transición. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrAvisoTransicion
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrAvisoTransicion

Salida

<i>TrAvisoTransicion[]</i>	Array de objetos TrAvisoTransicion
-----------------------------	------------------------------------

2.4.2.2.28 Mantenimiento de condiciones de documentos

2.4.2.2.28.1 Insertar condición de un documento

```
void insertarCondicionDocumentoPermitido (
    TrCondicionDocumentoPermitido condicionDocPer,
    TpoPK idTipoDoc, TpoPK idDefProc,
    TpoPK idFase, TpoPK idCondicion) throws TrException
```

Asocia una condición a un documento. Si todo es correcto devuelve los identificadores del tipo de documento, del procedimiento, de la fase y de la condición, si algo falla lanza una excepción.

Entradas

TrCondicionDocumentoPermitido Condición del documento a insertar
condicionDocPer

Entrada / Salida

<i>TpoPK</i> idTipoDoc	Identificador del tipo de documento o tarea, cero si no se ha insertado
<i>TpoPK</i> idDefProc	Identificador de la definición del procedimiento, cero si no se ha insertado
<i>TpoPK</i> idFase	Identificador de la fase, cero si no se ha insertado
<i>TpoPK</i> idCondicion	Identificador de la condición, cero si no se ha insertado

2.4.2.2.28.2 Modificar condición de un documento

int modificarCondicionDocumentoPermitido(

TrCondicionDocumentoPermitido condicionDocPer) throws TrException

Modifica una condición de un documento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrCondicionDocumentoPermitido Condición del documento a modificar
condicionDocPer

Salida

int Número de filas modificadas

2.4.2.2.28.3 Eliminar condición de un documento

int eliminarCondicionDocumentoPermitido(

TpoPK idTipoDoc,
TpoPK idDefProc,
TpoPK idFase,
TpoPK idCondicion) throws TrException

Elimina una condición de un documento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoDoc Identificador del tipo de documento o tarea

TpoPK idDefProc Identificador de la definición de procedimiento

TpoPK idFase Identificador de la fase

TpoPK idCondicion Identificador de la condición

Salida

int Número de filas eliminadas

2.4.2.2.28.4 Obtener condición de un documento

<p>TrCondicionDocumentoPermitido[] obtenerCondicionDocumentoPermitido(<i>TpoPK idTipoDoc,</i> <i>TpoPK idDefProc,</i> <i>TpoPK idFase,</i> <i>TpoPK idCondicion,</i> <i>ClausulaWhere where,</i> <i>ClausulaOrderBy orderBy)</i> throws <i>TrException</i></p>
--

Permite obtener una o varias condiciones de los documentos. Si no se pasa ningún parámetro se devuelven todas.

Entradas

<i>TpoPK idTipoDoc</i>	Identificador del tipo de documento o tarea
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento
<i>TpoPK idFase</i>	Identificador de la fase
<i>TpoPK idCondicion</i>	Identificador de la condición
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para <i>TrCondicionDocumentoPermitido</i>
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para <i>TrCondicionDocumentoPermitido</i>

Salida

<i>TrCondicionDocumentoPermitido[]</i>	Array de objetos <i>TrCondicionDocumentoPermitido</i>
---	---

2.4.2.2.29 Mantenimiento de acciones de documentos

2.4.2.2.29.1 Insertar acción de un documento

```
void insertarAccionDocumentoPermitido (
    TrAccionDocumentoPermitido accionDocPer,
    TpoPK idTipoDoc, TpoPK idDefProc,
    TpoPK idFase, TpoPK idAccion) throws TrException
```

Asocia una acción a un documento. Si todo es correcto devuelve los identificadores del tipo de documento, del procedimiento, de la fase y de la acción, si algo falla lanza una excepción.

Entradas

TrAccionDocumentoPermitido accionDocPer Acción del documento a insertar

Entrada / Salida

<i>TpoPK idTipoDoc</i>	Identificador del tipo de documento o tarea, cero si no se ha insertado
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento, cero si no se ha insertado
<i>TpoPK idFase</i>	Identificador de la fase, cero si no se ha insertado
<i>TpoPK idAccion</i>	Identificador de la acción, cero si no se ha insertado

2.4.2.2.29.2 Modificar acción de un documento

<p>int modificarAccionDocumentoPermitido(TrAccionDocumentoPermitido accionDocPer) throws TrException</p>
--

Modifica una acción de un documento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrAccionDocumentoPermitido accionDocPer Acción del documento a modificar

Salida

int Número de filas modificadas

2.4.2.2.29.3 Eliminar acción de un documento

<p>int eliminarAccionDocumentoPermitido(TpoPK idTipoDoc, TpoPK idDefProc, TpoPK idFase, TpoPK idAccion) throws TrException</p>
--

Elimina una acción de un documento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoDoc Identificador del tipo de documento o tarea

TpoPK idDefProc Identificador de la definición de procedimiento

TpoPK idFase Identificador de la fase

TpoPK idAccion Identificador de la acción

Salida

int Número de filas eliminadas

2.4.2.2.29.40 Obtener acción de un documento

<p>TrAccionDocumentoPermitido[] obtenerAccionDocumentoPermitido(TpoPK idTipoDoc, TpoPK idDefProc, TpoPK idFase, TpoPK idAccion, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Permite obtener una o varias acciones de los documentos. Si no se pasa ningún parámetro se devuelven todas.

Entradas

<i>TpoPK idTipoDoc</i>	Identificador del tipo de documento o tarea
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento
<i>TpoPK idFase</i>	Identificador de la fase
<i>TpoPK idAccion</i>	Identificador de la acción
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para <i>TrAccionDocumentoPermitido</i>
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para <i>TrAccionDocumentoPermitido</i>

Salida

<i>TrAccionDocumentoPermitido[]</i>	Array de objetos <i>TrAccionDocumentoPermitido</i>
--------------------------------------	--

2.4.2.2.30 Mantenimiento de metadatos

2.4.2.2.30.1 Insertar metadato

<p>TpoPK insertarAvisoDocumentoPermitido (TrMetadato metadato) throws TrException</p>

Registra un nuevo metadato para un sistema. Si todo es correcto devuelve el identificador del metadato, si algo falla lanza una excepción.

Entradas

TrMetadato metadato

Metadato a insertar

Entrada / Salida

TpoPK idMetadato

Identificador del metadato, cero si no se ha insertado

2.4.2.2.30.2 Modificar metadato

```
int modificarMetadato(
    TrMetadato metadato) throws TrException
```

Registra un nuevo metadato para un sistema. Si todo es correcto devuelve el identificador del metadato, si algo falla lanza una excepción.

Entradas

TrMetadato metadato

Metadato a modificar

Entrada / Salida

int metadato

Numero de registros modificados

2.4.2.2.30.3 Eliminar metadato

```
int eliminarMetadato(
    TpoPK idMetadato) throws TrException
```

Registra un nuevo metadato para un sistema. Si todo es correcto devuelve el identificador del metadato, si algo falla lanza una excepción.

Entradas

TpoPK idMetadato

Identificador del metadato a eliminar

Entrada / Salida

int Metadato

Numero de registros eliminados

2.4.2.2.30.4 Obtener metadato

```
TrMetadato[] obtenerMetadato(
    TpoPK idMetadato,
    ClausulaWhere where,
    ClausulaOrderBy orderBy) throws TrException
```

Registra un nuevo metadato para un sistema. Si todo es correcto devuelve el identificador del metadato, si algo falla lanza una excepción.

Entradas

TpoPK idMetadato

Identificador del metadato a obtener

Entrada / Salida

TrMetadato[] Metadatos

Array de metadatos obtenidos

2.4.2.2.31 Mantenimiento de avisos de documentos

2.4.2.2.31.1 Insertar aviso de un documento

<p>void insertarAvisoDocumentoPermitido (TrAvisoDocumentoPermitido avisoDocPer, TpoPK idTipoDoc, TpoPK idDefProc, TpoPK idFase, TpoPK idAviso) throws TrException</p>
--

Asocia un aviso a un documento. Si todo es correcto devuelve los identificadores del tipo de documento, del procedimiento, de la fase y del aviso, si algo falla lanza una excepción.

Entradas

TrAvisoDocumentoPermitido avisoDocPer	Aviso del documento a insertar
--	--------------------------------

Entrada / Salida

<i>TpoPK</i> idTipoDoc	Identificador del tipo de documento o tarea, cero si no se ha insertado
<i>TpoPK</i> idDefProc	Identificador de la definición del procedimiento, cero si no se ha insertado
<i>TpoPK</i> idFase	Identificador de la fase, cero si no se ha insertado
<i>TpoPK</i> idAviso	Identificador del aviso, cero si no se ha insertado

2.4.2.2.31.2 Modificar aviso de un documento

<p>int modificarAvisoDocumentoPermitido(TrAvisoDocumentoPermitido avisoDocPer) throws TrException</p>

Modifica un aviso de un documento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrAvisoDocumentoPermitido avisoDocPer	Aviso del documento a modificar
--	---------------------------------

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.31.3 Eliminar aviso de un documento

<p>int eliminarAvisoDocumentoPermitido(<i>TpoPK</i> idTipoDoc, <i>TpoPK</i> idDefProc, <i>TpoPK</i> idFase, <i>TpoPK</i> idAviso) throws TrException</p>
--

Elimina un aviso de un documento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idTipoDoc</i>	Identificador del tipo de documento o tarea
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento
<i>TpoPK idFase</i>	Identificador de la fase
<i>TpoPK idAviso</i>	Identificador del aviso

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.31.40 Obtener aviso de un documento

<p>TrAvisoDocumentoPermitido[] obtenerAvisoDocumentoPermitido(<i>TpoPK idTipoDoc,</i> <i>TpoPK idDefProc,</i> <i>TpoPK idFase,</i> <i>TpoPK idAviso,</i> <i>ClausulaWhere where,</i> <i>ClausulaOrderBy orderBy)</i> throws <i>TrException</i></p>
--

Permite obtener uno o varios avisos de los documentos. Si no se pasa ningún parámetro se devuelven todos.

Entradas

<i>TpoPK idTipoDoc</i>	Identificador del tipo de documento o tarea
<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento
<i>TpoPK idFase</i>	Identificador de la fase
<i>TpoPK idAviso</i>	Identificador del aviso
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para <i>TrAvisoDocumentoPermitido</i>
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para <i>TrAvisoDocumentoPermitido</i>

Salida*TrAvisoDocumentoPermitido[]*Array de objetos *TrAvisoDocumentoPermitido***2.4.2.2.32 Mantenimiento de condiciones de otras tareas****2.4.2.2.32.1 Insertar condición de otras tareas****void insertarCondicionBloquePermitido(**

```
TrCondicionBloquePermitido condicionBloquePer,  
TpoPK idCondicion,  
TpoPK idDefProc,  
TpoPK idBloquePer ) throws TrException
```

Asocia una condición a una tarea en fase o bloque permitido. Si todo es correcto devuelve los identificadores de la condición, de la definición del procedimiento y del bloque permitido, si algo falla lanza una excepción.

Entradas

TrCondicionBloquePermitido Condición del bloque permitido a insertar
condicionBloquePer

Entrada / Salida

TpoPK idCondicion Identificador de la condición, cero si no se ha insertado

TpoPK idDefProc Identificador de la definición del procedimiento, cero si no se ha insertado

TpoPK idBloquePer Identificador de la tarea en fase o bloque permitido, cero si no se ha insertado

2.4.2.2.32.2 Modificar condición de otras tareas

<p>int modificarCondicionBloquePermitido(<i>TrCondicionBloquePermitido condicionBloquePer</i>) throws <i>TrException</i></p>
--

Modifica una condición de un bloque permitido existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrCondicionBloquePermitido Condición del bloque permitido a modificar
condicionBloquePer

Salida

int Número de filas modificadas

2.4.2.2.32.3 Eliminar condición de otras tareas

<p>int eliminarCondicionBloquePermitido(<i>TpoPK idCondicion,</i> <i>TpoPK idDefProc,</i> <i>TpoPK idBloquePer</i>) throws <i>TrException</i></p>

Elimina una condición asociada a un bloque permitido existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idCondicion Identificador de la condición

TpoPK idDefProc Identificador de la definición del procedimiento

TpoPK idBloquePer Identificador del bloque permitido o tarea en fase

Salida

int Número de filas eliminadas

2.4.2.2.32.4 Obtener condición de otras tareas

```

TrCondicionBloquePermitido[ ] obtenerCondicionBloquePermitido(
    TpoPK idCondicion,
    TpoPK idDefProc,
    TpoPK idBloquePer,
    ClausulaWhere where,
    ClausulaOrderBy orderBy) throws TrException

```

Permite obtener una o varias condiciones asociadas a un bloque permitido. Si no se pasa ningún parámetro se devuelve todas.

Entradas

TpoPK idCondicion Identificador de la condición

TpoPK idDefProc Identificador de la definición del procedimiento

TpoPK idBloquePer Identificador del bloque permitido o tarea en fase

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrCondicionBloquePermitido

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrCondicionBloquePermitido

Salida

TrCondicionBloquePermitido[] Array de objetos TrCondicionBloquePermitido

2.4.2.2.33 Mantenimiento de acciones de otras tareas

2.4.2.2.33.1 Insertar acción de otras tareas

```

void insertarAccionBloquePermitido(
    TrAccionBloquePermitido accionBloquePer,
    TpoPK idAccion,
    TpoPK idDefProc,
    TpoPK idBloquePer ) throws TrException

```

Asocia una acción a una tarea en fase o bloque permitido. Si todo es correcto devuelve los identificadores de la acción, de la definición del procedimiento y del bloque permitido, si algo falla lanza una excepción.

Entradas

<i>TrAccionBloquePermitido</i>	Acción del bloque permitido a insertar
<i>accionBloquePer</i>	

Entrada / Salida

<i>TpoPK idAccion</i>	Identificador de la acción, cero si no se ha insertado
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento, cero si no se ha insertado
<i>TpoPK idBloquePer</i>	Identificador de la tarea en fase o bloque permitido, cero si no se ha insertado

2.4.2.2.33.2 Modificar acción de otras tareas

<p>int modificarAccionBloquePermitido(<i>TrAccionBloquePermitido accionBloquePer</i>) throws <i>TrException</i></p>
--

Modifica una acción de un bloque permitido existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrAccionBloquePermitido</i>	Acción del bloque permitido a modificar
<i>accionBloquePer</i>	

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.33.3 Eliminar acción de otras tareas

<p>int eliminarAccionBloquePermitido(<i>TpoPK idAccion,</i> <i>TpoPK idDefProc,</i> <i>TpoPK idBloquePer</i>) throws <i>TrException</i></p>

Elimina una acción asociada a un bloque permitido existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idAccion</i>	Identificador de la acción
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento
<i>TpoPK idBloquePer</i>	Identificador del bloque permitido o tarea en fase

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.33.4 Obtener acción de otras tareas

```

TrAccionBloquePermitido[ ] obtenerAccionBloquePermitido(
    TpoPK idAccion,
    TpoPK idDefProc,
    TpoPK idBloquePer,
    ClausulaWhere where,
    ClausulaOrderBy orderBy) throws TrException

```

Permite obtener una o varias acciones asociadas a un bloque permitido. Si no se pasa ningún parámetro se devuelve todas.

Entradas

<i>TpoPK idAccion</i>	Identificador de la acción
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento
<i>TpoPK idBloquePer</i>	Identificador del bloque permitido o tarea en fase
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrAccionBloquePermitido
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrAccionBloquePermitido

Salida

<i>TrAccionBloquePermitido[]</i>	Array de objetos TrAccionBloquePermitido
-----------------------------------	--

2.4.2.2.34 Mantenimiento de avisos de otras tareas

2.4.2.2.34.1 Insertar aviso de otras tareas

```

void insertarAvisoBloquePermitido(
    TrAvisoBloquePermitido avisoBloquePer,
    TpoPK idAviso,
    TpoPK idDefProc,
    TpoPK idBloquePer ) throws TrException

```

Asocia un aviso a una tarea en fase o bloque permitido. Si todo es correcto devuelve los identificadores del aviso, de la definición del procedimiento y del bloque permitido, si algo falla lanza una excepción.

Entradas

<i>TrAvisoBloquePermitido</i> <i>avisoBloquePer</i>	Aviso del bloque permitido a insertar
--	---------------------------------------

Entrada / Salida

<i>TpoPK idAviso</i>	Identificador del aviso, cero si no se ha insertado
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento, cero si no se ha insertado
<i>TpoPK idBloquePer</i>	Identificador de la tarea en fase o bloque permitido, cero si no se ha insertado

2.4.2.2.34.2 Modificar aviso de otras tareas

<p>int modificarAvisoBloquePermitido(<i>TrAvisoBloquePermitido</i> <i>avisoBloquePer</i>) throws <i>TrException</i></p>
--

Modifica un aviso de un bloque permitido existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrAvisoBloquePermitido</i> <i>avisoBloquePer</i>	Aviso del bloque permitido a modificar
--	--

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.34.3 Eliminar aviso de otras tareas

<p>int eliminarAvisoBloquePermitido(<i>TpoPK idAviso,</i> <i>TpoPK idDefProc,</i> <i>TpoPK idBloquePer</i>) throws <i>TrException</i></p>

Elimina un aviso asociado a un bloque permitido existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idAviso</i>	Identificador del aviso
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento
<i>TpoPK idBloquePer</i>	Identificador del bloque permitido o tarea en fase

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.34.4 Obtener aviso de otras tareas

```

TrAvisoBloquePermitido[ ] obtenerAvisoBloquePermitido(
    TpoPK idAviso,
    TpoPK idDefProc,
    TpoPK idBloquePer,
    ClausulaWhere where,
    ClausulaOrderBy orderBy) throws TrException

```

Permite obtener uno o varios avisos asociados a un bloque permitido. Si no se pasa ningún parámetro se devuelve todos.

Entradas

<i>TpoPK idAviso</i>	Identificador del aviso
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento
<i>TpoPK idBloquePer</i>	Identificador del bloque permitido o tarea en fase
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrAvisoBloquePermitido
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrAvisoBloquePermitido

Salida

<i>TrAvisoBloquePermitido[]</i>	Array de objetos TrAvisoBloquePermitido
----------------------------------	---

2.4.2.2.35 Mantenimiento de datos de un componente

2.4.2.2.35.1 Insertar datos de un componente

```

TpoPK insertarDatoComponente(
    TrDatoComponente datoComponente ) throws TrException

```

Inserta un nuevo dato de un componente. Si todo es correcto devuelve el identificador de la nueva dato de un componente, si algo falla lanza una excepción.

Entradas

<i>TrDatoComponente</i>	Dato de un componente a insertar
<i>datoComponente</i>	

Salida

<i>TpoPK</i>	Identificador del nuevo dato de un componente, cero si no se ha insertado
--------------	---

2.4.2.2.35.2 Modificar datos de un componente

```
int modificarDatoComponente(  

                                    TrDatoComponente datoComponente ) throws TrException
```

Modifica una dato de un componente existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrDatoComponente</i>	Dato de un componente a modificar
<i>datoComponente</i>	

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.35.3 Eliminar datos de un componente

```
int eliminarDatoComponente( TpoPK idDatoComponente) throws TrException
```

Elimina un dato de un componente existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idDatoComponente</i>	Identificador del dato de un componente a eliminar
-------------------------------	--

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.35.4 Obtener datos de un componente

```
TrDatoComponente [ ] obtenerDatoComponente( TpoPK idDatoComponente,  

                                                    ClausulaWhere where,  

                                                    ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener uno o varios datos de un componente. Si se pasa el parámetro *idDatoComponente* se devuelve los datos de esa dato de un componente, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK</i> idDatoComponente	Identificador del dato de un componente a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrDatoComponente
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrDatoComponente

Salida

<i>TrDatoComponente</i> []	Array de objetos TrDatoComponente
----------------------------	-----------------------------------

2.4.2.2.36 Mantenimiento de componentes

2.4.2.2.36.1 Insertar componente

TpoPK insertarComponente(TrComponente componente) throws TrException

Inserta un nuevo componente. Si todo es correcto devuelve el identificador de la nueva componente, si algo falla lanza una excepción.

Entradas

TrComponente componente	Componente a insertar
-------------------------	-----------------------

Salida

<i>TpoPK</i>	Identificador del nuevo componente, cero si no se ha insertado
--------------	--

2.4.2.2.36.2 Modificar componente

int modificarComponente(TrComponente componente) throws TrException

Modifica una componente existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrComponente componente Componente a modificar

Salida

int Número de filas modificadas

2.4.2.2.36.3 Eliminar componente

int eliminarComponente(TpoPK idComponente) throws TrException

Elimina un componente existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idComponente Identificador del componente a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.36.4 Obtener componente

TrComponente [] obtenerComponente(TpoPK idComponente, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios componentes. Si se pasa el parámetro *idComponente* se devuelve los datos de ese componente, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idComponente Identificador del componente a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrComponente

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrComponente

Salida

TrComponente[] Array de objetos TrComponente

2.4.2.2.37 Mantenimiento de constantes del sistema

2.4.2.2.37.1 Insertar constante

String insertarConstante(TrConstante constante) throws TrException

Inserta una nueva constante. Si todo es correcto devuelve el identificador de la nueva constante, si algo falla lanza una excepción.



Entradas

TrConstante constante Constante a insertar

Salida

String Identificador de la nueva constante, null si no se ha insertado

2.4.2.2.37.2 Modificar constante

int modificarConstante(TrConstante constante) throws TrException

Modifica una constante existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrConstante constante constante a modificar

Salida

int Número de filas modificadas

2.4.2.2.37.3 Eliminar constante

**int eliminarConstante(String idConstante
TpoPK idSistema) throws TrException**

Elimina una constante existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

String idConstante Identificador de la constante a eliminar

TpoPK idSistema Identificador del sistema

Salida

int Número de filas eliminadas

2.4.2.2.37.4 Obtener constante

**TrConstante [] obtenerConstante(String idConstante
TpoPK idSistema,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException**

Permite obtener una o varias constantes. Si se pasa el parámetro *idConstante* se devuelve los datos de esa constante, si se pasa *null*, se devuelven todas.

Entradas

<i>String</i> idConstante	Identificador de la constante a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK</i> idSistema	Identificador del sistema. Si se pasa <i>null</i> se obtienen las de todos los sistemas
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrConstante
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrConstante

Salida

<i>TrConstante</i> []	Array de objetos TrConstante
------------------------	------------------------------

2.4.2.2.38 Mantenimiento de tipos de actos

2.4.2.2.38.1 Insertar tipo de acto

TpoPK insertarTipoActo(TrTipoActo tipoActo) throws TrException

Inserta un nuevo tipo de acto. Si todo es correcto devuelve el identificador del nuevo tipo de acto, si algo falla lanza una excepción.

Entradas

<i>TrTipoActo</i> tipoActo	Tipo de acto a insertar
----------------------------	-------------------------

Salida

<i>TpoPK</i>	Identificador del nueva tipo de acto, cero si no se ha insertado
--------------	--

2.4.2.2.38.2 Modificar tipo de acto

int modificarTipoActo(TrTipoActo tipoActo) throws TrException

Modifica un tipo de acto existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoActo tipoActo Tipo de acto a modificar

Salida

int Número de filas modificadas

2.4.2.2.38.3 Eliminar tipo de acto

int eliminarTipoActo(TpoPK idTipoActo) throws TrException

Elimina un tipo de acto existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoActo Identificador del tipo de acto a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.38.4 Obtener tipo de acto

TrTipoActo [] obtenerTipoActo(TpoPK idTipoActo, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException
--

Permite obtener uno o varios tipos de actos. Si se pasa el parámetro *idTipoActo* se devuelve los datos de ese tipo de acto, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoActo Identificador del tipo de acto a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoActo

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoActo

Salida

TrTipoActo[] Array de objetos TrTipoActo

2.4.2.2.39 Mantenimiento de perfiles de usuario de una transición

2.4.2.2.39.1 Insertar perfil de usuario de una transición

void insertarTransicionPerfil (TrTransicionPerfil transicionPerfil,

TpoPK idDefProc,
TpoPK idTransicion,
TpoPK idPerfilUsu) throws TrException

Permite asociar un perfil de usuario a una transición. Si todo es correcto devuelve los identificadores de la transición, de la definición del procedimiento y del perfil de usuario, si algo falla lanza una excepción.

Entradas

<i>TrTransicionPerfil</i> transicionPerfil	Perfil y transición a asociar
---	-------------------------------

Entrada / Salida

<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento, cero si no se ha insertado
<i>TpoPK idTransicion</i>	Identificador de la transición, cero si no se ha insertado
<i>TpoPK idPerfilUsu</i>	Identificador del perfil de usuario, cero si no se ha insertado

2.4.2.2.39.2 Modificar perfil de usuario de una transición

int modificarTransicionPerfil(TrTransicionPerfil transicionPerfil) throws TrException

Modifica una relación entre una transición y un perfil de usuario existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrTransicionPerfil</i> transicionPerfil	Perfil y transición a modificar
---	---------------------------------

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.39.3 Eliminar perfil de usuario de una transición

**int eliminarTransicionPerfil(TpoPK idDefProc,
TpoPK idTransicion,
TpoPK idPerfilUsu) throws TrException**

Elimina una relación entre una transición y un perfil de usuario existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento
<i>TpoPK idTransicion</i>	Identificador de la transición
<i>TpoPK idPerfilUsu</i>	Identificador del perfil de usuario

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.39.4 Obtener perfil de usuario de una transición

TrTransicionPerfil[] obtenerTransicionPerfil(*TpoPK idDefProc,*
TpoPK idTransicion,
TpoPK idPerfilUsu,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws *TrException*

Permite obtener una o varias relaciones entre transiciones y perfiles de usuario. Si no se pasa ningún parámetro se devuelve todas y si se pasa alguno se filtra por él.

Entradas

<i>TpoPK idDefProc</i>	Identificador de la definición de procedimiento. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idTransicion</i>	Identificador de la transición. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idPerfilUsu</i>	Identificador del perfil de usuario. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para <i>TrTransicionPerfil</i>
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para <i>TrTransicionPerfil</i>

Salida

<i>TrTransicionPerfil[]</i>	Array de objetos <i>TrTransicionPerfil</i>
------------------------------	--

2.4.2.2.40 Mantenimiento de límites de caducidades

2.4.2.2.40.1 Insertar límite de caducidad

void insertarLimiteCaducidad (*TrLimiteCaducidad limiteCaducidad,*

```
TpoPK idCaducidad,  
TpoPK idTransicion) throws TrException;
```

Inserta un nuevo límite para una caducidad. Si todo es correcto devuelve el identificador de la caducidad y de la transición, si algo falla lanza una excepción.

Entradas

<i>TrLimiteCaducidad</i> limiteCaducidad	Límite para una caducidad a insertar
---	--------------------------------------

Entrada / Salida

<i>TpoPK idCaducidad</i>	Identificador de la caducidad, cero si no se ha insertado
<i>TpoPK idTransicion</i>	Identificador de la transición, cero si no se ha insertado

2.4.2.2.40.2 Eliminar límite de caducidad

```
int eliminarLimiteCaducidad( TpoPK idCaducidad,  
TpoPK idTransicion) throws TrException
```

Elimina un límite para una caducidad existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idCaducidad</i>	Identificador de la caducidad
<i>TpoPK idTransicion</i>	Identificador de la transición

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.40.3 Obtener límite de caducidad

```
TrLimiteCaducidad[ ] obtenerLimiteCaducidad( TpoPK idCaducidad,  
TpoPK idTransicion,  
ClausulaWhere where,  
ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener uno o varios límites para una caducidad. Si no se pasa ningún parámetros se devuelve todos los límites de caducidad de todas las transiciones y si se pasa algún parámetro se filtra el resultado por ese identificador.

Entradas

<i>TpoPK idCaducidad</i>	Identificador de la caducidad. Si se pasa <i>null</i> se obtienen todas.
<i>TpoPK idTransicion</i>	Identificador de la transición. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrLimiteCaducidad
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrLimiteCaducidad

Salida

<i>TrLimiteCaducidad[]</i>	Array de objetos TrLimiteCaducidad
-----------------------------	------------------------------------

2.4.2.2.41 Mantenimiento de documentos delegados

2.4.2.2.41.1 Insertar documento delegado

<p>TpoPK insertarDocumentoDelegado(TrDocumentoDelegado docDelegado) throws TrException</p>



Inserta un nuevo documento delegado. Si todo es correcto devuelve el identificador del nuevo documento delegado, si algo falla lanza una excepción.

Entradas

<i>TrDocumentoDelegado</i> <i>docDelegado</i>	Documento delegado a insertar
--	-------------------------------

Salida

<i>TpoPK</i>	Identificador del nuevo documento delegado, cero si no se ha insertado
--------------	--

2.4.2.2.41.2 Modificar documento delegado

int modificarDocumentoDelegado(TrDocumentoDelegado docDelegado) throws TrException

Modifica un documento delegado existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrDocumentoDelegado</i> <i>docDelegado</i>	Documento delegado a modificar
--	--------------------------------

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.41.3 Eliminar documento delegado

int eliminarDocumentoDelegado(TpoPK idDocDelegado) throws TrException

Elimina un documento delegado existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idDocDelegado</i>	Identificador del documento delegado a eliminar
----------------------------	---

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.41.4 Obtener documento delegado

TrDocumentoDelegado [] obtenerDocumentoDelegado(TpoPK idDocDelegado, ClausulaWhere where,

ClausulaOrderBy orderBy) throws TrException
--

Permite obtener uno o varios documentos delegados. Si se pasa el parámetro *idDocDelegado* se devuelve los datos de esa documento delegado, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK idDocDelegado</i>	Identificador del documento delegado a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrDocumentoDelegado
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrDocumentoDelegado

Salida

<i>TrDocumentoDelegado[]</i>	Array de objetos TrDocumentoDelegado
-------------------------------	--------------------------------------

2.4.2.2.42 Mantenimiento de ficha del procedimiento

2.4.2.2.42.1 Insertar ficha del procedimiento

TpoPK insertarIndicacionFicha(TrIndicacionFicha indicacionFicha) throws TrException

Inserta una nueva ficha para el procedimiento. Si todo es correcto devuelve el identificador de la nueva ficha para el procedimiento, si algo falla lanza una excepción.

Entradas

<i>TrIndicacionFicha</i> <i>indicacionFicha</i>	Ficha para el procedimiento a insertar
--	--

Salida

<i>TpoPK</i>	Identificador de la nueva ficha para el procedimiento, cero si no se ha insertado
--------------	---

2.4.2.2.42.2 Modificar ficha del procedimiento

int modificarIndicacionFicha(TrIndicacionFicha indicacionFicha) throws TrException

Modifica una ficha para el procedimiento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrIndicacionFicha</i> <i>indicacionFicha</i>	Ficha para el procedimiento a modificar
--	---

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.42.3 Eliminar ficha del procedimiento

int eliminarIndicacionFicha(TpoPK idIndFicha) throws TrException

Elimina una ficha para el procedimiento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idIndFicha</i>	Identificador de la ficha para el procedimiento a eliminar
-------------------------	--

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.42.4 Obtener ficha del procedimiento

TrIndicacionFicha [] obtenerIndicacionFicha(TpoPK idIndFicha, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias fichas del procedimiento. Si se pasa el parámetro *idIndFicha* se devuelve los datos de esa ficha para el procedimiento, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idIndFicha</i>	Identificador de la ficha para el procedimiento a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrIndicacionFicha
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrIndicacionFicha

Salida

<i>TrIndicacionFicha[]</i>	Array de objetos TrIndicacionFicha
-----------------------------	------------------------------------

2.4.2.2.43 Mantenimiento de normativas

2.4.2.2.43.1 Insertar normativa

TpoPK insertarNormativa (TrNormativa normativa) throws TrException
--

Inserta una nueva normativa. Si todo es correcto devuelve el identificador de la nueva normativa, si algo falla lanza una excepción.

Entradas

TrNormativa normativa Normativa a insertar

Salida

TpoPK Identificador de la nueva normativa, cero si no se ha insertado

2.4.2.2.43.2 Modificar normativa

int modificarNormativa(TrNormativa normativa) throws TrException

Modifica una normativa existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrNormativa normativa normativa a modificar

Salida

int Número de filas modificadas

2.4.2.2.43.3 Eliminar normativa

int eliminarNormativa(TpoPK idNormativa) throws TrException

Elimina una normativa existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idNormativa Identificador de la normativa a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.43.4 Obtener normativa

TrNormativa [] obtenerNormativa(TpoPK idNormativa, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias normativas. Si se pasa el parámetro *idNormativa* se devuelve los datos de esa normativa, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idNormativa</i>	Identificador de la normativa a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrNormativa
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrNormativa

Salida

<i>TrNormativa[]</i>	Array de objetos TrNormativa
-----------------------	------------------------------

2.4.2.2.44 Mantenimiento de normativas del procedimiento

2.4.2.2.44.1 Insertar normativa del procedimiento

```

void insertarNormativaDefProcedimiento(
    TrNormativaDefProcedimiento normativaDefProc,
    TpoPK idDefProc,
    TpoPK idNormativa) throws TrException

```

Permite asociar una normativa a un procedimiento. Si todo es correcto los identificadores de la normativa y de la definición del procedimiento, si algo falla lanza una excepción.

Entradas

TrNormativaDefProcedimiento	Nomativa y procedimiento a relacionar
normativaDefProc	

Entrada / Salida

<i>TpoPK</i> idDefProc	Identificador de la definición del procedimiento, cero si no se ha insertado
<i>TpoPK</i> idNormativa	Identificador de la normativa, cero si no se ha insertado

2.4.2.2.44.2 Eliminar normativa del procedimiento

int eliminarNormativaDefProcedimiento(<i>TpoPK</i> idDefProc, <i>TpoPK</i> idNormativa) throws TrException
--

Elimina una normativa asociada a un procedimiento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK</i> idDefProc	Identificador de la definición del procedimiento
<i>TpoPK</i> idNormativa	Identificador de la normativa

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.44.3 Eliminar normativas del procedimiento

int eliminarNormativaDefProcedimiento(<i>TpoPK</i> idDefProc) throws TrException

Elimina todas las normativas asociadas a un procedimiento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idDefProc

Identificador de la definición del procedimiento

Salida

int

Número de filas eliminadas

2.4.2.2.44.4 Obtener normativa del procedimiento

TrNormativaDefProcedimiento[] obtenerNormativaDefProcedimiento(

TpoPK idDefProc,
TpoPK idNormativa,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias normativas asociadas a algún procedimiento. Si no se pasa ningún parámetro se obtienen todas las normativas de todos los procedimientos y si se pasa alguno de los identificadores se filtrará por él.

Entradas

TpoPK idDefProc

Identificador de la definición del procedimiento. Si se pasa *null* se obtienen todas.

TpoPK idNormativa

Identificador de la normativa. Si se pasa *null* se obtienen todas.

ClausulaWhere where

Filtro a aplicar.
Consultar campos posibles para TrNormativaDefProcedimiento

ClausulaOrderBy orderBy

Ordenación a aplicar.
Consultar campos posibles para TrNormativaDefProcedimiento

Salida

TrNormativaDefProcedimiento[]

Array de objetos TrNormativaDefProcedimiento

2.4.2.2.45 Mantenimiento de organismos

2.4.2.2.45.1 Insertar organismo

TpoPK insertarOrganismo(TrOrganismo organismo) throws TrException

Inserta un nuevo organismo. Si todo es correcto devuelve el identificador del nuevo organismo, si algo falla lanza una excepción.

Entradas

TrOrganismo organismo Organismo a insertar

Salida

TpoPK Identificador del nuevo organismo, cero si no se ha insertado

2.4.2.2.45.2 Modificar organismo

int modificarOrganismo(TrOrganismo organismo) throws TrException

Modifica un organismo existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrOrganismo organismo Organismo a modificar

Salida

int Número de filas modificadas

2.4.2.2.45.3 Eliminar organismo

int eliminarOrganismo(TpoPK idOrganismo) throws TrException

Elimina un organismo existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idOrganismo Identificador del organismo a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.45.4 Obtener organismo

TrOrganismo [] obtenerOrganismo(TpoPK idOrganismo, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios organismos. Si se pasa el parámetro *idOrganismo* se devuelve los datos de ese organismo, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK</i> idOrganismo	Identificador del organismo a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrOrganismo
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrOrganismo

Salida

<i>TrOrganismo</i> []	Array de objetos TrOrganismo
-----------------------	------------------------------

2.4.2.2.46 Mantenimiento de organismos del procedimiento

2.4.2.2.46.1 Insertar organismo del procedimiento

TpoPK insertarOrganismoProcedimiento(TrOrganismoProcedimiento orgDefProc) throws TrException

Inserta un nuevo organismo del procedimiento. Si todo es correcto devuelve el identificador de la nueva relación, si algo falla lanza una excepción.

Entradas

TrOrganismoProcedimiento Organismo del procedimiento a insertar
orgDefProc

Salida

TpoPK Identificador de la nueva relación entre el organismo y el procedimiento.

2.4.2.2.46.2 Modificar organismo del procedimiento

int modificarOrganismoProcedimiento(<i>TrOrganismoProcedimiento</i> <i>orgDefProc</i>) throws <i>TrException</i>

Modifica un organismo del procedimiento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrOrganismoProcedimiento Organismo del procedimiento a modificar
orgDefProc

Salida

int Número de filas modificadas

2.4.2.2.46.3 Eliminar organismo del procedimiento

int eliminarOrganismoProcedimiento(<i>TpoPK</i> <i>idOrgProc</i>) throws <i>TrException</i>

Elimina una relación entre organismo y procedimiento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK *idOrgProc* Identificador de la relación entre el organismo y el procedimiento.

Salida

int Número de filas eliminadas

2.4.2.2.46.4 Obtener organismo

TrOrganismoProcedimiento[] obtenerOrganismosProcedimiento(<i>TpoPK</i> <i>idOrgProc</i> <i>boolean</i> <i>incluirOrgProc</i> <i>ClausulaWhere</i> <i>where</i> ,
--

ClausulaOrderBy orderBy) throws TrException
--

Devuelve una lista de organismos asociados al procedimiento.

Entradas

<i>TpoPK idOrgProc</i>	<i>Identificador de la relación entre el organismo y el procedimiento</i>
<i>boolean incluirOrgProc</i>	<i>Indica si se incluyen en el conjunto los organismos pertenece, competente, tramita y resuelve que ya existían en la entidad de procedimientos.</i> <i>Si se incluyen, en TEXTOASOCIA se indica el organismo pertenece mediante la cadena PERTENECE, el organismo competente mediante la cadena COMPETENTE, el organismo resuelve mediante la cadena RESUELVE y el organismo tramita mediante la cadena TRAMITA. Para estos organismos del procedimiento, los demás atributos como REFORGPROC, OBSERVACIONES, y fechas de vigencia irán nulas.</i>
<i>ClausulaWhere where</i>	<i>Filtro a aplicar.</i> <i>Consultar campos posibles para TrOrganismoProcedimiento.</i>
<i>ClausulaOrderBy orderBy</i>	<i>Ordenación a aplicar.</i> <i>Consultar campos posibles para TrOrganismoProcedimiento.</i>

Salida

TrOrganismoProcedimiento [] Array de objetos TrOrganismoProcedimiento.

2.4.2.2.47 Mantenimiento de plantillas del procedimiento

2.4.2.2.47.1 Insertar plantilla del procedimiento

TpoPK insertarPlantillaProcedimiento(TrPlantillaProcedimiento plantillaProc) throws TrException

Inserta una nueva plantilla del procedimiento. Si todo es correcto devuelve el identificador de la nueva plantilla del procedimiento, si algo falla lanza una excepción.

Entradas

TrPlantillaProcedimiento plantillaProc	Plantilla del procedimiento a insertar
---	--

Salida

TpoPK	Identificador de la nueva plantilla del procedimiento, cero si no se ha insertado
-------	---

2.4.2.2.47.2 Modificar plantilla del procedimiento

int modificarPlantillaProcedimiento(TrPlantillaProcedimiento plantillaProc) throws TrException

Modifica una plantilla del procedimiento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrPlantillaProcedimiento plantillaProc	plantilla del procedimiento a modificar
---	---

Salida

int	Número de filas modificadas
-----	-----------------------------

2.4.2.2.47.3 Eliminar plantilla del procedimiento

int eliminarPlantillaProcedimiento(TpoPK idPlantillaProc) throws TrException

Elimina una plantilla del procedimiento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idPlantillaProc	Identificador de la plantilla del procedimiento a eliminar
-----------------------	--

Salida

int	Número de filas eliminadas
-----	----------------------------

2.4.2.2.47.4 Obtener plantilla del procedimiento

TrPlantillaProcedimiento [] obtenerPlantillaProcedimiento(TpoPK idPlantillaProc,

ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias plantillas del procedimiento. Si se pasa el parámetro *idPlantillaProc* se devuelve los datos de esa plantilla del procedimiento, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idPlantillaProc</i>	Identificador de la plantilla del procedimiento a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrPlantillaProcedimiento
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrPlantillaProcedimiento

Salida

<i>TrPlantillaProcedimiento[]</i>	Array de objetos TrPlantillaProcedimiento
------------------------------------	---

2.4.2.2.48 Mantenimiento de tipo certificado del procedimiento

2.4.2.2.48.1 Insertar tipo certificado del procedimiento

void insertarTipoCertificadoDefProcedimiento (TrTipoCertificadoProcedimiento tipoCertDefProc, TpoPK idDefProc, TpoPK idTipoCert) throws TrException

Inserta una nueva tipo certificado del procedimiento. Si todo es correcto devuelve el identificador del procedimiento y del tipo de certificado asociados, si algo falla lanza una excepción.

Entradas

<i>TrTipoCertificadoProcedimiento</i> tipoCertDefProc	Tipo certificado del procedimiento a insertar
--	---

Salida

<i>TpoPK idDefProc</i>	Identificador del procedimiento
<i>TpoPK idTipoCert</i>	Identificador del tipo certificado

2.4.2.2.48.2 Modificar tipo certificado del procedimiento

int modificarTipoCertificadoDefProcedimiento (TrTipoCertificadoProcedimiento tipoCertDefProc) throws TrException

Modifica un tipo certificado del procedimiento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrTipoCertificadoProcedimiento</i> tipoCertDefProc	Tipo certificado del procedimiento a modificar
--	--

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.48.3 Eliminar tipo certificado del procedimiento

int eliminarTipoCertificadoDefProcedimiento (TpoPK idDefProc, TpoPK idTipoCert) throws TrException

Elimina un tip certificado del procedimiento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idDefProc</i>	Identificador del procedimiento
<i>TpoPK idTipoCert</i>	Identificador del tipo de certificado a eliminar

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.48.4 Eliminar tipo certificado del procedimiento

int eliminarTipoCertificadoDefProcedimiento (TpoPK idDefProc) throws TrException

Elimina todos los tipos de certificados del procedimiento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idDefProc Identificador del procedimiento

Salida

int Número de filas eliminadas

2.4.2.2.48.5 Obtener tipo certificado del procedimiento

TrTipoCertificadoProcedimiento [] obtenerPlantillaProcedimiento(TpoPK idDefProc, TpoPK idTipoCert, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener los tipos certificados asociados a procedimientos.

Entradas

TpoPK idDefProc Identificador del procedimiento. Si se pasa *null* se obtienen todas.

TpoPK idTipoCert Identificador del tipo certificado. Si se pasa *null* se obtienen todas.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoCertificadoProcedimiento

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoCertificadoProcedimiento

Salida

TrTipoCertificadoProcedimiento [] Array de objetos TrTipoCertificadoProcedimiento

2.4.2.2.49 Mantenimiento de errores



2.4.2.2.49.1 Insertar error

TpoPK insertarError(TrError error) throws TrException

Inserta un nuevo error. Si todo es correcto devuelve el identificador de la nueva error, si algo falla lanza una excepción.

Entradas

TrError error	Error a insertar
---------------	------------------

Salida

TpoPK	Identificador del nuevo error, cero si no se ha insertado
-------	---

2.4.2.2.49.2 Modificar error

int modificarError(TrError error) throws TrException

Modifica un error existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrError error	Error a modificar
---------------	-------------------

Salida

int	Número de filas modificadas
-----	-----------------------------

2.4.2.2.49.3 Eliminar error

int eliminarError(TpoPK idError) throws TrException

Elimina un error existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idError	Identificador del error a eliminar
---------------	------------------------------------

Salida

int	Número de filas eliminadas
-----	----------------------------

2.4.2.2.49.4 Obtener error

TrError [] obtenerError(TpoPK idError,

ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios errores Si se pasa el parámetro *idError* se devuelve los datos de ese error, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK idError</i>	Identificador del error a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrError
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrError

Salida

<i>TrError[]</i>	Array de objetos TrError
-------------------	--------------------------

2.4.2.2.50 Mantenimiento de firmantes definidos

2.4.2.2.50.1 Insertar firmante definido

TpoPK insertarFirmanteDefinido(TrFirmanteDefinido firmanteDefinido) throws TrException

Inserta un nuevo firmante definido. Si todo es correcto devuelve el identificador de la nueva firmante definido, si algo falla lanza una excepción.

Entradas

TrFirmanteDefinido firmanteDefinido	Firmante definido a insertar
--	------------------------------

Salida

TpoPK	Identificador del nuevo firmante definido, cero si no se ha insertado
-------	---

2.4.2.2.50.2 Modificar firmante definido

int modificarFirmanteDefinido(TrFirmanteDefinido firmanteDefinido) throws TrException

Modifica un firmante definido existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrFirmanteDefinido firmanteDefinido	Firmante definido a modificar
--	-------------------------------

Salida

int	Número de filas modificadas
-----	-----------------------------

2.4.2.2.50.3 Eliminar firmante definido

int eliminarFirmanteDefinido(TpoPK idFirmanteDef) throws TrException

Elimina un firmante definido existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idFirmanteDef	Identificador del firmante definido a eliminar
---------------------	--

Salida

int	Número de filas eliminadas
-----	----------------------------

2.4.2.2.50.4 Obtener firmante definido

TrFirmanteDefinido [] obtenerFirmanteDefinido(TpoPK idFirmanteDef,

ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios firmantes definidos. Si se pasa el parámetro *idFirmanteDef* se devuelve los datos de ese firmante definido, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK idFirmanteDef</i>	Identificador del firmante definido a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrFirmanteDefinido
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrFirmanteDefinido

Salida

<i>TrFirmanteDefinido[]</i>	Array de objetos TrFirmanteDefinido
------------------------------	-------------------------------------

2.4.2.2.51 Mantenimiento de firmas

2.4.2.2.51.1 Insertar firma

TpoPK insertarFirma(TrFirma firma) throws TrException

Inserta una nueva firma. Si todo es correcto devuelve el identificador de la nueva firma, si algo falla lanza una excepción.

Entradas

TrFirma firma	Firma a insertar
---------------	------------------

Salida

TpoPK	Identificador de la nueva firma, cero si no se ha insertado
-------	---

2.4.2.2.51.2 Modificar firma

int modificarFirma(TrFirma firma) throws TrException

Modifica una firma existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrFirma firma	Firma a modificar
---------------	-------------------

Salida

int	Número de filas modificadas
-----	-----------------------------

2.4.2.2.51.3 Eliminar firma

int eliminarFirma(TpoPK idFirma) throws TrException

Elimina una firma existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idFirma	Identificador de la firma a eliminar
---------------	--------------------------------------

Salida

int	Número de filas eliminadas
-----	----------------------------

2.4.2.2.51.4 Obtener firma

TrFirma [] obtenerFirma(TpoPK idFirma, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias firmas. Si se pasa el parámetro *idFirma* se devuelve los datos de esa firma, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idFirma</i>	Identificador de la firma a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrFirma
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrFirma

Salida

<i>TrFirma[]</i>	Array de objetos TrFirma
-------------------	--------------------------

2.4.2.2.52 Mantenimiento de plantillas

2.4.2.2.52.1 Insertar plantilla

TpoPK insertarPlantilla(TrPlantilla plantilla) throws TrException

Inserta una nueva plantilla para la generación de documentos. Si todo es correcto devuelve el identificador de la nueva plantilla, si algo falla lanza una excepción.

Entradas

TrPlantilla plantilla Plantilla a insertar

Salida

TpoPK Identificador de la nueva plantilla, cero si no se ha insertado

2.4.2.2.52.2 Modificar plantilla

int modificarPlantilla(TrPlantilla plantilla) throws TrException

Modifica una plantilla para la generación de documentos existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrPlantilla plantilla Plantilla a modificar

Salida

int Número de filas modificadas

2.4.2.2.52.3 Eliminar plantilla

int eliminarPlantilla(TpoPK idPlantilla) throws TrException

Elimina una plantilla para la generación de documentos existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idPlantilla Identificador de la plantilla a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.52.4 Obtener plantilla

TrPlantilla [] obtenerPlantilla(TpoPK idPlantilla, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias plantillas para la generación de documentos. Si se pasa el parámetro *idPlantilla* se devuelve los datos de esa plantilla, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK</i> idPlantilla	Identificador de la plantilla a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrPlantilla
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrPlantilla

Salida

<i>TrPlantilla</i> []	Array de objetos TrPlantilla
------------------------	------------------------------

2.4.2.2.53 Mantenimiento de relaciones

2.4.2.2.53.1 Insertar relación

TpoPK insertarRelacion(TrRelacion relacion) throws TrException

Inserta una nueva relación entre fases, transiciones, tipos de documentos, etc. Si todo es correcto devuelve el identificador de la nueva relación, si algo falla lanza una excepción.

Entradas

TrRelacion relacion relación a insertar

Salida

TpoPK Identificador de la nueva relación, cero si no se ha insertado

2.4.2.2.53.2 Modificar relación

int modificarRelacion(TrRelacion relacion) throws TrException

Modifica una relación existente entre fases, transiciones, tipos de documentos, etc. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrRelacion relacion relación a modificar

Salida

int Número de filas modificadas

2.4.2.2.53.3 Eliminar relación

int eliminarRelacion(TpoPK idRelacion) throws TrException

Elimina una relación existente entre fases, transiciones, tipos de documentos, etc. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idRelacion Identificador de la relación a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.53.4 Obtener relación

TrRelacion [] obtenerRelacion(TpoPK idRelacion, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias relaciones entre fases, transiciones, tipos de documentos, etc. Si se pasa el parámetro *idRelacion* se devuelve los datos de esa relación, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idRelacion</i>	Identificador de la relación a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrRelacion
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrRelacion

Salida

<i>TrRelacion[]</i>	Array de objetos TrRelacion
----------------------	-----------------------------

2.4.2.2.54 Mantenimiento de disposiciones de firma

2.4.2.2.54.1 Insertar disposición para la firma

<p>TpoPK insertarTextoDisposicionFirma(TrTextoDisposicionFirma textoDisp) throws TrException</p>
--

Inserta una nueva disposición para la firma. Si todo es correcto devuelve el identificador de la nueva disposición para la firma, si algo falla lanza una excepción.



Entradas

TrTextoDisposicionFirma Disposición para la firma a insertar
textoDisp

Salida

TpoPK Identificador de la nueva disposición para la firma, cero si no se ha insertado

2.4.2.2.54.2 Modificar disposición para la firma

int modificarTextoDisposicionFirma(TrTextoDisposicionFirma textoDisp) throws TrException

Modifica una disposición para la firma existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTextoDisposicionFirma Disposición para la firma a modificar
textoDisp

Salida

int Número de filas modificadas

2.4.2.2.54.3 Eliminar disposición para la firma

int eliminarTextoDisposicionFirma(TpoPK idTextoDisp) throws TrException

Elimina una disposición para la firma existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTextoDisp Identificador de la disposición para la firma a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.54.4 Obtener disposición para la firma

TrTextoDisposicionFirma [] obtenerTextoDisposicionFirma(TpoPK idTextoDisp, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias disposiciones para las firmas. Si se pasa el parámetro *idTextoDisp* se devuelve los datos de esa disposición para la firma, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idTextoDisp</i>	Identificador de la disposición para la firma a obtener. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para <i>TrTextoDisposicionFirma</i>
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para <i>TrTextoDisposicionFirma</i>

Salida

<i>TrTextoDisposicionFirma[]</i>	Array de objetos <i>TrTextoDisposicionFirma</i>
-----------------------------------	---

2.4.2.2.55 Mantenimiento de parámetros

2.4.2.2.55.1 Insertar parámetro

TpoPK insertarParametro (<i>TrParametro parametro</i>) throws <i>TrException</i>

Inserta un nuevo parámetro. Si todo es correcto devuelve el identificador del nuevo parámetro, si algo falla lanza una excepción.



Entradas

TrParametro parametro Parámetro a insertar

Salida

TpoPK Identificador del nuevo parámetro, cero si no se ha insertado

2.4.2.2.55.2 Modificar parámetro

int modificarParametro(TrParametro parametro) throws TrException

Modifica un parámetro existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrParametro parametro Parámetro a modificar

Salida

int Número de filas modificadas

2.4.2.2.55.3 Eliminar parámetro

int eliminarParametro(TpoPK idParametro) throws TrException

Elimina un parámetro existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idParametro Identificador del parámetro a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.55.4 Obtener parámetro

TrParametro [] obtenerParametro(TpoPK idParametro, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios parámetros. Si se pasa el parámetro *idParametro* se devuelve los datos de ese parámetro, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK</i> idParametro	Identificador del parámetro a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrParametro
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrParametro

Salida

<i>TrParametro</i> []	Array de objetos TrParametro
------------------------	------------------------------

2.4.2.2.56 Mantenimiento de párrafos

2.4.2.2.56.1 Insertar párrafo

TpoPK insertarParrafoTipoDocumento(TrParrafoTipoDocumento parrafoTipoDoc) throws TrException

Inserta un nuevo párrafo para un tipo de documento. Si todo es correcto devuelve el identificador del nuevo párrafo, si algo falla lanza una excepción.



Entradas

TrParrafoTipoDocumento Párrafo a insertar
parrafoTipoDoc

Salida

TpoPK Identificador del nuevo párrafo, cero si no se ha insertado

2.4.2.2.56.2 Modificar párrafo

int modificarParrafoTipoDocumento(*TrParrafoTipoDocumento* *parrafoTipoDoc*) throws *TrException*

Modifica un párrafo para un tipo de documento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrParrafoTipoDocumento Párrafo a modificar
parrafoTipoDoc

Salida

int Número de filas modificadas

2.4.2.2.56.3 Eliminar párrafo

int eliminarParrafoTipoDocumento(*TpoPK* *idParrafoTipoDoc*) throws *TrException*

Elimina un párrafo para un tipo de documento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idParrafoTipoDoc Identificador del párrafo a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.56.4 Obtener párrafo

TrParrafoTipoDocumento [] obtenerParrafoTipoDocumento(
 TpoPK idParrafoTipoDoc,
 ClausulaWhere where,
 ClausulaOrderBy orderBy) throws *TrException*

Permite obtener uno o varios párrafos. Si se pasa el parámetro *idParrafoTipoDoc* se devuelve los datos de ese párrafo, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK idParrafoTipoDoc</i>	Identificador del párrafo a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrParrafoTipoDocumento
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrParrafoTipoDocumento

Salida

<i>TrParrafoTipoDocumento[]</i>	Array de objetos TrParrafoTipoDocumento
---------------------------------	---

2.4.2.2.57 Mantenimiento de tipos de párrafos

2.4.2.2.57.1 Insertar tipo de párrafo

TpoPK insertarTipoParrafo (TrTipoParrafo tipoParrafo) throws TrException
--

Inserta un nuevo tipo de párrafo. Si todo es correcto devuelve el identificador del nuevo tipo de párrafo, si algo falla lanza una excepción.

Entradas

TrTipoParrafo tipoParrafo Tipo de párrafo a insertar

Salida

TpoPK Identificador del nuevo tipo de párrafo, cero si no se ha insertado

2.4.2.2.57.2 Modificar tipo de párrafo

```
int modificarTipoParrafo( TrTipoParrafo tipoParrafo ) throws TrException
```

Modifica un tipo de párrafo existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoParrafo tipoParrafo Tipo de párrafo a modificar

Salida

int Número de filas modificadas

2.4.2.2.57.3 Eliminar tipo de párrafo

```
int eliminarTipoParrafo( TpoPK idTipoParrafo) throws TrException
```

Elimina un tipo de párrafo existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoParrafo Identificador del tipo de párrafo a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.57.4 Obtener tipo de párrafo

```
TrTipoParrafo [ ] obtenerTipoParrafo( TpoPK idTipoParrafo,  
ClausulaWhere where,  
ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener uno o varios tipos de párrafos. Si se pasa el parámetro *idTipoParrafo* se devuelve los datos de ese tipo de párrafo, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoParrafo Identificador del tipo de párrafo a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoParrafo

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoParrafo

Salida

TrTipoParrafo[] Array de objetos TrTipoParrafo

2.4.2.2.58 Mantenimiento de tipos de relaciones

2.4.2.2.58.1 Insertar tipo de relación

TpoPK insertarTipoRelacion(TrTipoRelacion tipoRelacion) throws TrException

Inserta un nuevo tipo de relación. Si todo es correcto devuelve el identificador del nuevo tipo de relación, si algo falla lanza una excepción.

Entradas

TrTipoRelacion tipoRelacion Tipo de relación a insertar

Salida

TpoPK Identificador del nuevo tipo de relación, cero si no se ha insertado

2.4.2.2.58.2 Modificar tipo de relación

int modificarTipoRelacion(TrTipoRelacion tipoRelacion) throws TrException

Modifica un tipo de relación existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoRelacion tipoRelacion Tipo de relación a modificar

Salida

int Número de filas modificadas

2.4.2.2.58.3 Eliminar tipo de relación

int eliminarTipoRelacion(TpoPK idTipoRelacion) throws TrException

Elimina un tipo de relación existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoRelacion Identificador del tipo de relación a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.58.4 Obtener tipo de relación

TrTipoRelacion [] obtenerTipoRelacion(TpoPK idTipoRelacion, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios tipos de relación. Si se pasa el parámetro *idTipoRelacion* se devuelve los datos de ese tipo de relación, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoRelacion Identificador del tipo de relación a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoRelacion

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoRelacion

Salida

TrTipoRelacion[] Array de objetos TrTipoRelacion

2.4.2.2.59 Mantenimiento de variables

2.4.2.2.59.1 Insertar variable

TpoPK insertarVariable(TrVariable variable) throws TrException

Inserta una nueva variable. Si todo es correcto devuelve el identificador de la nueva variable, si algo falla lanza una excepción.

Entradas

TrVariable variable

Variable a insertar

Salida

TpoPK

Identificador de la nueva variable, cero si no se ha insertado

2.4.2.2.59.2 Modificar variable

int modificarVariable(TrVariable variable) throws TrException

Modifica una variable existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrVariable variable

Variable a modificar

Salida

int

Número de filas modificadas

2.4.2.2.59.3 Eliminar variable

int eliminarVariable(TpoPK idVariable) throws TrException

Elimina una variable existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idVariable

Identificador de la variable a eliminar

Salida

int

Número de filas eliminadas

2.4.2.2.59.4 Obtener variable

**TrVariable [] obtenerVariable(TpoPK idVariable,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException**

Permite obtener una o varias variables, Si se pasa el parámetro *idVariable* se devuelve los datos de esa variable, si se pasa *null*, se devuelven todas.

Entradas

TpoPK idVariable Identificador de la variable a obtener. Si se pasa *null* se obtienen todas.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrVariable

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrVariable

Salida

TrVariable[] Array de objetos TrVariable

2.4.2.2.60 Mantenimiento de perfiles de usuario de otras tareas

2.4.2.2.60.1 Insertar perfil de usuario de otras tareas

<p>TpoPK insertarBloquePermitidoPerfil(TrBloquePermitidoPerfil blq, TpoPK idPerfilUsu, TpoPK idDefProc, TpoPK idBloquePer) throws TrException</p>
--

Permite asociar un perfil de usuario a una tarea. Si todo es correcto devuelve los identificadores de la tarea, de la definición del procedimiento y del perfil de usuario, si algo falla lanza una excepción.

Entradas

TrBloquePermitidoPerfil blq Perfil y tarea a asociar.

Entrada / Salida

TpoPK idDefProc Identificador de la definición del procedimiento, cero si no se ha insertado

TpoPK idBloquePer Identificador de la tarea, cero si no se ha insertado

TpoPK idPerfilUsu Identificador del perfil de usuario, cero si no se ha insertado

2.4.2.2.60.2 Eliminar perfil de usuario de otras tareas

int eliminarBloquePermitidoPerfil(*TpoPK idPerfilUsu,*
TpoPK idDefProc,
TpoPK idBloquePer) throws *TrException*

Elimina la relación entre un perfil de usuario y una tarea en fase existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idPerfilUsu Identificador del perfil de usuario.

TpoPK idDefProc Identificador de la definición del procedimiento.

TpoPK idBloquePer Identificador de la tarea en fase o bloque permitido.

Salida

int Número de filas eliminadas

2.4.2.2.60.3 Obtener perfil de usuario de otras tareas

TrBloquePermitidoPerfil[] obtenerBloquePermitidoPerfil(*TpoPK idPerfilUsu,*
TpoPK idDefProc,
TpoPK idBloquePer,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws *TrException*

Permite obtener uno o varios perfiles asociados a las tareas en fase. Si no se pasan ningún parámetro se devuelven todos los perfiles, si se pasa alguno se filtra por él.

Entradas

<i>TpoPK idPerfilUsu</i>	Identificador del perfil de usuario.
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento.
<i>TpoPK idBloquePer</i>	Identificador de la tarea en fase o bloque permitido.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrBloquePermitidoPerfil
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrBloquePermitidoPerfil

Salida

<i>TrBloquePermitidoPerfil[]</i>	Array de objetos TrBloquePermitidoPerfil
-----------------------------------	--

2.4.2.2.61 Mantenimiento de perfiles de usuario de documentos

2.4.2.2.61.1 Insertar perfil de usuario de un documento

```

void insertarDocumentoPermitidoPerfil( TrDocumentoPermitidoPerfil doc,
                                         TpoString permiso,
                                         TpoPK idPerfilUsu,
                                         TpoPK idTipoDoc,
                                         TpoPK idFase,
                                         TpoPK idDefProc) throws TrException

```

Permite asociar un perfil de usuario a un documento. Si todo es correcto devuelve los identificadores de la tarea o tipo de documento, de la definición del procedimiento, del perfil de usuario, de la fase y

el permiso, si algo falla lanza una excepción. Los valores para el permiso puede ser “G” para Generar, “I” para Incorporar, “F” para Firmar, “E” para Editar y “T” para todo.

Entradas

TrDocumentoPermitidoPerfil doc Perfil y documento a asociar.

Entrada / Salida

<i>TpoString</i> permiso	Permiso para el perfil. Sus valores pueden ser: → “ G ” – Generar → “ I ” – Incorporar → “ F ” – Firmar → “ E ” – Editar → “ T ” – Todo
<i>TpoPK idPerfilUsu</i>	Identificador del perfil de usuario, cero si no se ha insertado
<i>TpoPK idTipoDoc</i>	Identificador de la tarea o tipo de documento, cero si no se ha insertado
<i>TpoPK idFase</i>	Identificador de la fase, cero si no se ha insertado
<i>TpoPK idDefProc</i>	Identificador de la definición del procedimiento, cero si no se ha insertado

2.4.2.2.61.2 Eliminar perfil de usuario de un documento

```
int eliminarDocumentoPermitidoPerfil( String permiso,
                                     TpoPK idPerfilUsu,
                                     TpoPK idTipoDoc,
                                     TpoPK idFase,
                                     TpoPK idDefProc ) throws TrException
```

Elimina la relación entre un perfil de usuario y una tarea en fase existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>String</i> permiso	Identificador del perfil de usuario.
<i>TpoPK</i> idPerfilUsu	Identificador del perfil de usuario.
<i>TpoPK</i> idTipoDoc	Identificador de la tarea o tipo de documento.
<i>TpoPK</i> idFase	Identificador de la fase.
<i>TpoPK</i> idDefProc	Identificador de la definición del procedimiento.

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.61.3 Obtener perfil de usuario de otras tareas

<p>TrDocumentoPermitidoPerfil[] obtenerDocumentoPermitidoPerfil(String permiso , TpoPK idPerfilUsu, TpoPK idTipoDoc, TpoPK idFase, TpoPK idDefProc, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Permite obtener uno o varios perfiles asociados a las tareas en fase o documentos. Si no se pasan ningún parámetro se devuelven todos los perfiles, si se pasa alguno se filtra por él.

Entradas

<i>String</i> permiso	Identificador del perfil de usuario.
<i>TpoPK</i> idPerfilUsu	Identificador del perfil de usuario.
<i>TpoPK</i> idTipoDoc	Identificador de la tarea o tipo de documento.
<i>TpoPK</i> idFase	Identificador de la fase.
<i>TpoPK</i> idDefProc	Identificador de la definición del procedimiento.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrDocumentoPermitidoPerfil
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrDocumentoPermitidoPerfil

Salida

TrDocumentoPermitidoPerfil[] Array de objetos TrDocumentoPermitidoPerfil

2.4.2.2.62 Mantenimiento de firmas de tipos de documentos

2.4.2.2.62.1 Insertar firma de tipo de documento

```
void insertarFirmaTipoDocumento(TrFirmaTipoDocumento firmaTipoDoc,  
                                TpoPK idTipoDoc,  
                                TpoPK idFirmanteDef) throws TrException
```

Inserta una nueva firma para un tipo de documento. Si todo es correcto devuelve los identificadores del tipo de documento y del firmante definido, si algo falla lanza una excepción.

Entradas

TrFirmaTipoDocumento Firma para un tipo de documento a insertar
firmaTipoDoc

Entrada / Salida

<i>TpoPK idTipoDoc</i>	Identificador del tipo de documento, cero si no se ha insertado
<i>TpoPK idFirmanteDef</i>	Identificador del firmante definido, cero si no se ha insertado

2.4.2.2.62.2 Modificar firma de tipo de documento

int modificarFirmaTipoDocumento(TrFirmaTipoDocumento firmaTipoDoc) throws TrException

Modifica una firma para un tipo de documento existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrFirmaTipoDocumento</i> <i>firmaTipoDoc</i>	Firma para un tipo de documento a modificar
--	---

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.62.3 Eliminar firma de tipo de documento

int eliminarFirmaTipoDocumento(TpoPK idTipoDoc, TpoPK idFirmanteDef) throws TrException

Elimina una firma para un tipo de documento existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>TpoPK idTipoDoc</i>	Identificador del tipo de documento
<i>TpoPK idFirmanteDef</i>	Identificador del firmante definido

Salida

int Número de filas eliminadas

2.4.2.2.62.4 Obtener firma de tipo de documento

```
TrFirmaTipoDocumento[ ] obtenerFirmaTipoDocumento( TpoPK idTipoDoc,
TpoPK idFirmanteDef,
ClausulaWhere where,
ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener una o varias firmas para un tipo de documento. Si se pasa el parámetro *idTipoDoc* se devuelve los datos de las firmas para ese tipo de documento y si se pasa el *idFirmanteDef* se filtra por ese firmante, si se pasa *null*, se devuelven todas.

Entradas

<i>TpoPK idTipoDoc</i>	Identificador del tipo de documento. Si se pasa <i>null</i> se obtienen todos.
<i>TpoPK idFirmanteDef</i>	Identificador del firmante definido. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrFirmaTipoDocumento
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrFirmaTipoDocumento

Salida

TrFirmaTipoDocumento[] Array de objetos TrFirmaTipoDocumento

2.4.2.2.63 Mantenimiento de parámetros de otras tareas

2.4.2.2.63.1 Insertar parámetro de otras tareas

```
void insertarParametroBloque(TrParametroBloque paramBloque,
TpoPK idParametro,
TpoPK idBloque) throws TrException
```

Inserta un nuevo parámetro para otras tareas. Si todo es correcto devuelve el identificador del parámetro y de la tarea o bloque, si algo falla lanza una excepción.

Entradas

TrParametroBloque
paramBloque

Parámetro para otras tareas a insertar

Entrada / Salida

TpoPK idParametro

Identificador del parámetro, cero si no se ha insertado

TpoPK idBloque

Identificador del bloque o tarea, cero si no se ha insertado

2.4.2.2.63.2 Modificar parámetro de otras tareas

int modificarParametroBloque(TrParametroBloque paramBloque) throws TrException

Modifica un parámetro para otras tareas existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrParametroBloque
paramBloque

Parámetro para otras tareas a modificar

Salida

int

Número de filas modificadas

2.4.2.2.63.3 Eliminar parámetro de otras tareas

**int eliminarParametroBloque(TpoPK idParametro,
TpoPK idBloque)** throws TrException

Elimina un parámetro para otras tareas existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idParametro

Identificador del parámetro

TpoPK idBloque Identificador del bloque o tarea

Salida

int Número de filas eliminadas

2.4.2.2.63.4 Obtener parámetro de otras tareas

```
TrParametroBloque[ ] obtenerParametroBloque( TpoPK idParametro,  
TpoPK idBloque,  
ClausulaWhere where,  
ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener uno o varios parámetros asociados a otras tareas. Si se pasa el parámetro *idParametro* se obtienen los datos de ese parámetro para otras tareas y si se pasa el parámetro *idBloque* se obtienen los parámetros para ese bloque o tarea, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idParametro Identificador del parámetro

TpoPK idBloque Identificador del bloque o tarea

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrParametroBloque

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrParametroBloque

Salida

TrParametroBloque[] Array de objetos TrParametroBloque

2.4.2.2.64 Mantenimiento de parámetros de variables

2.4.2.2.64.1 Insertar parámetro de variable

```
void insertarParametroVariable(TrParametroVariable paramVar,  
TpoPK idParametro,  
TpoPK idVariable) throws TrException
```

Inserta un nuevo parámetro para las variables. Si todo es correcto devuelve el identificador del parámetro y de la variable, si algo falla lanza una excepción.

Entradas

TrParametroVariable paramVar Parámetro para las variables a insertar

Entrada / Salida

TpoPK idParametro Identificador del parámetro, cero si no se ha insertado

TpoPK idVariable Identificador de la variable, cero si no se ha insertado

2.4.2.2.64.2 Modificar parámetro de variable

int modificarParametroVariable(TrParametroVariable paramVar) throws TrException

Modifica un parámetro para una variable existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrParametroVariable paramVar Parámetro para la variable a modificar

Salida

int Número de filas modificadas

2.4.2.2.64.3 Eliminar parámetro de variable

int eliminarParametroVariable(TpoPK idParametro, TpoPK idVariable) throws TrException
--

Elimina un parámetro para una variable existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idParametro Identificador del parámetro

TpoPK idVariable Identificador de la variable

Salida

int Número de filas eliminadas

2.4.2.2.64.4 Obtener parámetro de variable

TrParametroVariable[] obtenerParametroVariable(*TpoPK idParametro,*
TpoPK idVariable,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws *TrException*

Permite obtener uno o varios parámetros asociados a variable. Si se pasa el parámetro *idParametro* se obtienen los datos de ese parámetro para las variables y si se pasa el parámetro *idVariable* se obtienen los parámetros de esa variable, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idParametro Identificador del parámetro

TpoPK idVariable Identificador de la variable

ClausulaWhere where Filtro a aplicar.
 Consultar campos posibles para *TrParametroVariable*

ClausulaOrderBy orderBy Ordenación a aplicar.
 Consultar campos posibles para *TrParametroVariable*

Salida

TrParametroVariable[] Array de objetos *TrParametroVariable*

2.4.2.2.65 Mantenimiento de perfiles de usuario de un usuario

2.4.2.2.65.1 Insertar perfil de usuario de un usuario

void insertarUsuarioPerfilUsuario (*TrUsuarioPerfilUsuario usuPer,*
TpoPK idPerfilUsu,
TpoString usuario) throws *TrException*

Permite asignar un perfil de usuario a un usuario. Si todo es correcto devuelve los identificadores del perfil de usuario y del usuario, si algo falla lanza una excepción.

Entradas

TrUsuarioPerfilUsuario usuPer Usuario y perfil de usuario a relacionar

Entrada / Salida

TpoPK idPerfilUsu Identificador del perfil de usuario, cero si no se ha insertado

TpoString usuario Identificador del usuario, cero si no se ha insertado

2.4.2.2.65.2 Eliminar perfil de usuario de un usuario

**int eliminarUsuarioPerfilUsuario(TpoPK idPerfilUsu,
String usuario) throws TrException**

Elimina una relación existente entre un perfil de usuario y un usuario. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idPerfilUsu Identificador del perfil de usuario

String usuario Identificador del usuario

Salida

int Número de filas eliminadas

2.4.2.2.65.3 Obtener perfil de usuario de un usuario

**TrUsuarioPerfilUsuario[] obtenerUsuarioPerfilUsuario(
TpoPK idPerfilUsu,
String usuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException**

Permite obtener uno o varios perfiles de usuario que tiene asignados un usuario. Si se pasa el parámetro *idPerfilUsu* se devuelve los datos de ese perfil de usuario para todos los usuarios, si se pasa el parámetro *usuario* se devuelven los perfiles de ese usuario y si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK idPerfilUsu</i>	Identificador del perfil de usuario. Si se pasa <i>null</i> se obtienen todos.
<i>String usuario</i>	Identificador del usuario. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrUsuarioPerfilUsuario
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrUsuarioPerfilUsuario

Salida

<i>TrUsuarioPerfilUsuario[]</i>	Array de objetos TrUsuarioPerfilUsuario
---------------------------------	---

2.4.2.2.66 Mantenimiento de ámbitos de ley

2.4.2.2.66.1 Insertar ámbito de ley

TpoPK insertarAmbitoLey (TrAmbitoLey ambitoLey) throws TrException
--

Inserta un nuevo ámbito de ley. Si todo es correcto devuelve el identificador del nuevo ámbito de ley, si algo falla lanza una excepción.

Entradas

TrAmbitoLey ambitoLey Ámbito de ley a insertar

Salida

TpoPK Identificador del nuevo ámbito de ley, cero si no se ha insertado

2.4.2.2.66.2 Modificar ámbito de ley

int modificarAmbitoLey(TrAmbitoLey ambitoLey) throws TrException

Modifica un ámbito de ley existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrAmbitoLey ambitoLey Ámbito de ley a modificar

Salida

int Número de filas modificadas

2.4.2.2.66.3 Eliminar ámbito de ley

int eliminarAmbitoLey(TpoPK idAmbitoLey) throws TrException

Elimina un ámbito de ley existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idAmbitoLey Identificador del ámbito de ley a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.66.4 Obtener ámbito de ley

TrAmbitoLey [] obtenerAmbitoLey(TpoPK idAmbitoLey, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios ámbitos de ley. Si se pasa el parámetro *idAmbitoLey* se devuelve los datos de ese ámbito de ley, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK</i> idAmbitoLey	Identificador del ámbito de ley a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrAmbitoLey
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrAmbitoLey

Salida

<i>TrAmbitoLey</i> []	Array de objetos TrAmbitoLey
------------------------	------------------------------

2.4.2.2.67 Mantenimiento de razones de interés

2.4.2.2.67.1 Insertar razón de interés

TpoPK insertarRazonInteres(TrRazonInteres razonInt) throws TrException

Inserta una nueva razón de interés. Si todo es correcto devuelve el identificador de la nueva razón de interés, si algo falla lanza una excepción.

Entradas

TrRazonInteres razonInt Razón de interés a insertar

Salida

TpoPK Identificador de la nueva razón de interés, cero si no se ha insertado

2.4.2.2.67.2 Modificar razón de interés

int modificarRazonInteres(TrRazonInteres razonInt) throws TrException

Modifica una razón de interés existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrRazonInteres razonInt Razón de interés a modificar

Salida

int Número de filas modificadas

2.4.2.2.67.3 Eliminar razón de interés

int eliminarRazonInteres(TpoPK idRazonInt) throws TrException

Elimina una razón de interés existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idRazonInt Identificador de la razón de interés a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.67.4 Obtener razón de interés

TrRazonInteres [] obtenerRazonInteres(TpoPK idRazonInt, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias razones de interés. Si se pasa el parámetro *idRazonInt* se devuelve los datos de esa razón de interés, si se pasa *null*, se devuelven todas.

Entradas

TpoPK idRazonInt Identificador de la razón de interés a obtener. Si se pasa *null* se obtienen todas.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrRazonInteres

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrRazonInteres

Salida

TrRazonInteres[] Array de objetos TrRazonInteres

2.4.2.2.68 Mantenimiento de tipos de pago

2.4.2.2.68.1 Insertar tipo de pago

TpoPK insertarTipoPago(TrTipoPago tipoPago) throws TrException

Inserta un nuevo tipo de pago. Si todo es correcto devuelve el identificador del nuevo tipo de pago, si algo falla lanza una excepción.

Entradas

TrTipoPago tipoPago

Tipo de pago a insertar

Salida

TpoPK

Identificador del nuevo tipo de pago, cero si no se ha insertado

2.4.2.2.68.2 Modificar tipo de pago

TpoPK modificarTipoPago(TrTipoPago tipoPago) throws TrException

Modifica un tipo de pago existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoPago tipoPago

Tipo de pago a modificar

Salida

int

Número de filas modificadas

2.4.2.2.68.3 Eliminar tipo de pago

TpoPK eliminarTipoPago(TrTipoPago tipoPago) throws TrException

Elimina un tipo de pago existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TrTipoPago tipoPago

Tipo de pago a eliminar

Salida

int

Número de filas eliminadas

2.4.2.2.68.4 Obtener tipo de pago

TrTipoPago [] obtenerTipoPago(TpoPK idTipoPago, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios tipos de pagos. Si se pasa el parámetro *idTipoPago* se devuelve los datos de ese tipo de pago, si se pasa *null*, se devuelven todas.

Entradas

TpoPK idTipoPago Identificador del tipo de pago a obtener. Si se pasa *null* se obtienen todas.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoPago

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoPago

Salida

TrRazonInteres[] Array de objetos TrTipoPago

2.4.2.2.69 Mantenimiento de tipos de componentes

2.4.2.2.69.1 Insertar tipo de componente

TpoPK insertarTipoComponente(TrTipoComponente tipoComp) throws TrException

Inserta un nuevo tipo de componente. Si todo es correcto devuelve el identificador del nuevo tipo de componente, si algo falla lanza una excepción.

Entradas

TrTipoComponente tipoComp Tipo de componente a insertar

Salida

TpoPK Identificador del nuevo tipo de componente, cero si no se ha insertado

2.4.2.2.69.2 Modificar tipo de componente

int modificarTipoComponente(TrTipoComponente tipoComp) throws TrException

Modifica un tipo de componente existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoComponente tipoComp Tipo de componente a modificar

Salida

int Número de filas modificadas

2.4.2.2.69.3 Eliminar tipo de componente

int eliminarTipoComponente(TpoPK idTipoComp) throws TrException

Elimina un tipo de componente existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoComp Identificador del tipo de componente a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.69.4 Obtener tipo de componente

TrTipoComponente [] obtenerTipoComponente(TpoPK idTipoComp, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios tipos de componentes. Si se pasa el parámetro *idTipoComp* se devuelve los datos de ese tipo de componente, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoComp Identificador del tipo de componente a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoComponente

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoComponente

Salida

TrTipoComponente[] Array de objetos TrTipoComponente

2.4.2.2.70 Mantenimiento de tipos de contacto

2.4.2.2.70.1 Insertar tipo de contacto

TpoPK insertarTipoContacto(TrTipoContacto tipoCont) throws TrException

Inserta un nuevo tipo de contacto. Si todo es correcto devuelve el identificador de la nueva tipo de contacto, si algo falla lanza una excepción.

Entradas

TrTipoContacto tipoCont Tipo de contacto a insertar

Salida

TpoPK Identificador del nuevo tipo de contacto, cero si no se ha insertado

2.4.2.2.70.2 Modificar tipo de contacto

int modificarTipoContacto(TrTipoContacto tipoCont) throws TrException

Modifica un tipo de contacto existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoContacto tipoCont Tipo de contacto a modificar

Salida

int Número de filas modificadas

2.4.2.2.70.3 Eliminar tipo de contacto

int eliminarTipoContacto(TpoPK idTipoCont) throws TrException

Elimina un tipo de contacto existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoCont Identificador del tipo de contacto a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.70.4 Obtener tipo de contacto

TrTipoContacto [] obtenerTipoContacto(TpoPK idTipoCont, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios tipos de contacto. Si se pasa el parámetro *idTipoCont* se devuelve los datos de ese tipo de contacto, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoCont Identificador del tipo de contacto a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoContacto

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoContacto

Salida

TrTipoContacto[] Array de objetos TrTipoContacto

2.4.2.2.71 Mantenimiento de tipos de indicaciones

2.4.2.2.71.1 Insertar tipo de indicación

TpoPK insertarTipoIndicacion(TrTipoIndicacion tipoInd) throws TrException

Inserta un nuevo tipo de indicación. Si todo es correcto devuelve el identificador del nuevo tipo de indicación, si algo falla lanza una excepción.

Entradas

TrTipoIndicacion tipoInd Tipo de indicación a insertar

Salida

TpoPK Identificador del nuevo tipo de indicación, cero si no se ha insertado

2.4.2.2.71.2 Modificar tipo de indicación

int modificarTipoIndicacion(TrTipoIndicacion tipoInd) throws TrException

Modifica un tipo de indicación existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoIndicacion tipoInd Tipo de indicación a modificar

Salida

int Número de filas modificadas

2.4.2.2.71.3 Eliminar tipo de indicación

int eliminarTipoIndicacion(TpoPK idTipoInd) throws TrException

Elimina un tipo de indicación existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoInd Identificador del tipo de indicación a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.71.4 Obtener tipo de indicación

TrTipoIndicacion [] obtenerTipoIndicacion(TpoPK idTipoInd, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios tipos de indicación. Si se pasa el parámetro *idTipoInd* se devuelve los datos de ese tipo de indicación, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoInd Identificador del tipo de indicación a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoIndicacion

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoIndicacion

Salida

TrTipoIndicacion[] Array de objetos TrTipoIndicacion

2.4.2.2.72 Mantenimiento de tipos de normativas

2.4.2.2.72.1 Insertar tipo de normativa

TpoPK insertarTipoNormativa(TrTipoNormativa tipoNorma) throws TrException

Inserta un nuevo tipo de normativa. Si todo es correcto devuelve el identificador del nuevo tipo de normativa, si algo falla lanza una excepción.

Entradas

TrTipoNormativa tipoNorma Tipo de normativa a insertar

Salida

TpoPK Identificador del nuevo tipo de normativa, cero si no se ha insertado

2.4.2.2.72.2 Modificar tipo de normativa

int modificarTipoNormativa(TrTipoNormativa tipoNorma) throws TrException

Modifica un tipo de normativa existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoNormativa tipoNorma Tipo de normativa a modificar

Salida

int Número de filas modificadas

2.4.2.2.72.3 Eliminar tipo de normativa

int eliminarTipoNormativa(TpoPK idTipoNorma) throws TrException

Elimina un tipo de normativa existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoNorma Identificador del tipo de normativa a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.72.4 Obtener tipo de normativa

TrTipoNormativa [] obtenerTipoNormativa(TpoPK idTipoNorma, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios tipos de normativas. Si se pasa el parámetro *idTipoNorma* se devuelve los datos de ese tipo de normativa, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoNorma Identificador del tipo de normativa a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoNormativa

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoNormativa

Salida

TrTipoNormativa[] Array de objetos TrTipoNormativa

2.4.2.2.73 Mantenimiento de tipos de publicación

2.4.2.2.73.1 Insertar tipo de publicación

TpoPK insertarTipoPublicacion(TrTipoPublicacion tipoPubli) throws TrException

Inserta un nuevo tipo de publicación. Si todo es correcto devuelve el identificador del nuevo tipo de publicación, si algo falla lanza una excepción.

Entradas

TrTipoPublicacion tipoPubli Tipo de publicación a insertar

Salida

TpoPK Identificador del nuevo tipo de publicación, cero si no se ha insertado

2.4.2.2.73.2 Modificar tipo de publicación

int modificarTipoPublicacion(TrTipoPublicacion tipoPubli) throws TrException

Modifica un tipo de publicación existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoPublicacion tipoPubli Tipo de publicación a modificar

Salida

int Número de filas modificadas

2.4.2.2.73.3 Eliminar tipo de publicación

int eliminarTipoPublicacion(TpoPK idTipoPubli) throws TrException

Elimina un tipo de publicación existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoPubli Identificador del tipo de publicación a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.73.4 Obtener tipo de publicación

TrTipoPublicacion [] obtenerTipoPublicacion(TpoPK idTipoPubli, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios tipos de publicación. Si se pasa el parámetro *idTipoPubli* se devuelve los datos de ese tipo de publicación, si se pasa *null*, se devuelven todos.

Entradas

<i>TpoPK idTipoPubli</i>	Identificador del tipo de publicación a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrTipoPublicacion
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrTipoPublicacion

Salida

<i>TrTipoPublicacion[]</i>	Array de objetos TrTipoPublicacion
-----------------------------	------------------------------------

2.4.2.2.74 Mantenimiento de variables para un tipo de documento

2.4.2.2.74.1 Insertar variable para un tipo de documento

```
void insertarVariableTipoDocumento( TrVariableTipoDocumento variable,
                                     TpoPK idTipoDoc,
                                     TpoPK idVariable) throws TrException
```

Permite relacionar una variable con un tipo de documento. Si todo es correcto devuelve los identificadores del tipo de documento y de la variable, si algo falla lanza una excepción.

Entradas

TrVariableTipoDocumento variable Variable y tipo de documento a relacionar.

Entrada / Salida

TpoPK idTipoDoc Identificador del tipo de documento, cero si no se ha insertado

TpoPK idVariable Identificador de la variable, cero si no se ha insertado

2.4.2.2.74.2 Eliminar variable para un tipo de documento

<p>int eliminarVariableTipoDocumento(TpoPK idTipoDoc, TpoPK idVariable) throws TrException</p>
--

Elimina una relación existente entre una variable y un tipo de documento. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoDoc Identificador del tipo de documento

TpoPK idVariable Identificador de la variable

Salida

int Número de filas eliminadas

2.4.2.2.74.3 Eliminar todas las variables para un tipo de documento

<p>int eliminarVariableTipoDocumento(TpoPK idTipoDoc) throws TrException</p>
--

Permite eliminar todas las variables asociadas a tipo de documento que se indique en el parámetro *idTipoDoc*. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoDoc Identificador del tipo de documento

TrConstanteGeneral
constanteGn Constante general a insertar

Salida

String Identificador de la nueva constante general

2.4.2.2.75.2 Modificar constante general

int modificarConstanteGeneral(TrConstanteGeneral constanteGn) throws TrException

Modifica una constante general existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrConstanteGeneral
constanteGn Constante general a modificar

Salida

int Número de filas modificadas

2.4.2.2.75.3 Eliminar constante general

int eliminarConstanteGeneral(TpoPK idConstanteGn) throws TrException

Elimina una constante general existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idConstanteGn Identificador de la constante general a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.75.4 Obtener constante general

TrConstanteGeneral [] obtenerConstanteGeneral(TpoPK idConstanteGn, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias constantes generales. Si se pasa el parámetro *idConstanteGn* se devuelve los datos de esa constante general, si se pasa *null*, se devuelven todas.

Entradas

TpoPK idConstanteGn Identificador de la constante general a obtener. Si se pasa *null* se obtienen todas.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrConstanteGeneral

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrConstanteGeneral

Salida

TrConstanteGeneral[] Array de objetos TrConstanteGeneral

2.4.2.2.76 Mantenimiento de empleados

2.4.2.2.76.1 Insertar empleado

```
void insertarEmpleado( TrEmpleado empleado,
                      TpoString usuario,
                      TpoPK idOrganismo,
                      TpoString idPuestoTrab) throws TrException
```

Inserta un nuevo empleado. Si todo es correcto devuelve el identificador del nuevo empleado, si algo falla lanza una excepción.

Entradas

<i>TrEmpleado</i> empleado	Empleado a insertar
----------------------------	---------------------

Entrada / Salida

<i>TpoString</i> usuario	Identificador del usuario
<i>TpoPK</i> idOrganismo	Identificador del organismo
<i>TpoString</i> idPuestoTrab	Identificador del puesto de trabajo

2.4.2.2.76.2 Modificar empleado

int modificarEmpleado(TrEmpleado empleado) throws TrException
--

Modifica un empleado existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrEmpleado</i> empleado	Empleado a modificar
----------------------------	----------------------

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.76.3 Eliminar empleado

int eliminarEmpleado(String usuario, TpoPK idOrganismo, String idPuestoTrab) throws TrException
--

Elimina un empleado existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>String</i> usuario	Identificador del usuario
-----------------------	---------------------------

TpoPK idOrganismo Identificador del organismo

String idPuestoTrab Identificador del puesto de trabajo

Salida

int Número de filas eliminadas

2.4.2.2.76.4 Obtener empleado

<p>TrEmpleado[] obtenerEmpleado(String usuario, TpoPK idOrganismo, String idPuestoTrab, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>
--

Permite obtener uno o varios empleados. Si se pasa alguno de los parámetros se filtra el resultado, si se pasan todos a *null*, se devuelven todos.

Entradas

String usuario Identificador del usuario, si se pasa *null* se obtienen todos.

TpoPK idOrganismo Identificador del organismo, si se pasa *null* se obtienen todos.

String idPuestoTrab Identificador del puesto de trabajo, si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrEmpleado

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrEmpleado

Salida

TrEmpleado[] Array de objetos TrEmpleado

<p>int obtenerEmpleadoCount(String usuario, TpoPK idOrganismo, String idPuestoTrab, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException</p>

Devuelve el número de empleados

Entradas

<i>String</i> usuario	Identificador del usuario, si se pasa <i>null</i> se obtienen todos.
<i>TpoPK</i> idOrganismo	Identificador del organismo, si se pasa <i>null</i> se obtienen todos.
<i>String</i> idPuestoTrab	Identificador del puesto de trabajo, si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrEmpleado
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrEmpleado

Salida

<i>int</i>	Número de registros
------------	---------------------

2.4.2.2.77 Mantenimiento de provincias

2.4.2.2.77.1 Insertar provincia

String insertarProvincia(TrProvincia provincia) throws TrException

Inserta una nueva provincia. Si todo es correcto devuelve el identificador de la nueva provincia, si algo falla lanza una excepción.

Entradas

TrProvincia provincia	Provincia a insertar
-----------------------	----------------------

Salida

String Identificador de la nueva provincia

2.4.2.2.77.2 Modificar provincia

int modificarProvincia(TrProvincia provincia) throws TrException

Modifica una provincia existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrProvincia provincia Provincia a modificar

Salida

int Número de filas modificadas

2.4.2.2.77.3 Eliminar provincia

int eliminarProvincia(String idProvincia) throws TrException

Elimina una provincia existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

String idProvincia Identificador de la provincia a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.77.4 Obtener provincia

TrProvincia [] obtenerProvincia(String idProvincia, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener una o varias provincias. Si se pasa el parámetro *idProvincia* se devuelve los datos de esa provincia, si se pasa *null*, se devuelven todas.

Entradas

String idProvincia Identificador de la provincia a obtener. Si se pasa *null* se obtienen todas.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrProvincia

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrProvincia

Salida

TrProvincia[] Array de objetos TrProvincia

2.4.2.2.78 Mantenimiento de municipios

2.4.2.2.78.1 Insertar municipio

<pre>void insertarMunicipio(TrMunicipio municipio, TpoString idMunicipio, TpoString idProvincia) throws TrException</pre>
--

Inserta un nuevo municipio. Si todo es correcto devuelve los identificadores del municipio y provincia, si algo falla lanza una excepción.

Entradas

TrMunicipio municipio Municipio a insertar

Salida

<i>TpoString</i> idMunicipio	Identificador del municipio
<i>TpoString</i> idProvincia	Identificador de la provincia

2.4.2.2.78.2 Modificar municipio

int modificarMunicipio(TrMunicipio municipio) throws TrException

Modifica un municipio existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

<i>TrMunicipio</i> municipio	Municipio a modificar
------------------------------	-----------------------

Salida

<i>int</i>	Número de filas modificadas
------------	-----------------------------

2.4.2.2.78.3 Eliminar municipio

**int eliminarMunicipio(String idMunicipio,
String idProvincia) throws TrException**

Elimina un municipio existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>String</i> idMunicipio	Identificador del municipio
<i>String</i> idProvincia	Identificador de la provincia

Salida

int Número de filas eliminadas

2.4.2.2.78.4 Obtener municipio

TrMunicipio[] obtenerMunicipio(String idMunicipio,
String idProvincia,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios municipios. Si se pasa algunos de los parámetros *idMunicipio* o *idProvincia* el resultado se filtra, si se pasan ambos *null*, se devuelven todos.

Entradas

<i>String</i> idMunicipio	Identificador del municipio. Si se pasa <i>null</i> se obtienen todas.
<i>String</i> idProvincia	Identificador de la provincia. Si se pasa <i>null</i> se obtienen todas.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrMunicipio
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrMunicipio

Salida

TrMunicipio[] Array de objetos TrMunicipio

2.4.2.2.79 Mantenimiento de países

2.4.2.2.79.1 Insertar país

String insertarPais(TrPais pais) throws TrException

Inserta un nuevo país. Si todo es correcto devuelve el identificador del nuevo país, si algo falla lanza una excepción.

Entradas

TrPais pais	País a insertar
-------------	-----------------

Salida

String

Identificador del nuevo país

2.4.2.2.79.2 Modificar país

int modificarPais(TrPais pais) throws TrException

Modifica un país existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrPais pais

País a modificar

Salida

int

Número de filas modificadas

2.4.2.2.79.3 Eliminar país

int eliminarPais(String idPais) throws TrException

Elimina un país existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

String idPais

Identificador del país a eliminar

Salida

int

Número de filas eliminadas

2.4.2.2.79.4 Obtener país

TrPais [] obtenerPais(String idPais, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios países. Si se pasa el parámetro *idPais* se devuelve los datos de ese país, si se pasa *null*, se devuelven todas.

Entradas

String idPais

Identificador del país a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrPais

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrPais

Salida

TrPais[] Array de objetos TrPais

2.4.2.2.80 Mantenimiento de puestos de trabajo

2.4.2.2.80.1 Insertar puesto de trabajo

String insertarPuestoTrabajo(TrPuestoTrabajo puestoTrabajo) throws TrException

Inserta un nuevo puesto de trabajo. Si todo es correcto devuelve el identificador del nuevo puesto de trabajo, si algo falla lanza una excepción.

Entradas

TrPuestoTrabajo puestoTrabajo Puesto de trabajo a insertar

Salida

String Identificador del nuevo puesto de trabajo.

2.4.2.2.80.2 Modificar puesto de trabajo

int modificarPuestoTrabajo(TrPuestoTrabajo puestoTrabajo) throws TrException

Modifica un puesto de trabajo existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrPuestoTrabajo puestoTrabajo Puesto de trabajo a modificar

Salida

int Número de filas modificadas

2.4.2.2.80.3 Eliminar puesto de trabajo

int eliminarPuestoTrabajo(String idPuestoTrab) throws TrException

Elimina un puesto de trabajo existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idPuestoTrab Identificador del puesto de trabajo a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.80.4 Obtener puesto de trabajo

TrPuestoTrabajo [] obtenerPuestoTrabajo(String idPuestoTrab, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios puestos de trabajo. Si se pasa el parámetro *idPuestoTrab* se devuelven los datos de ese puesto de trabajo, si se pasa *null*, se devuelven todos.

Entradas

String idPuestoTrab Identificador del puesto de trabajo a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrPuestoTrabajo

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrPuestoTrabajo

Salida

TrPuestoTrabajo[] Array de objetos TrPuestoTrabajo

2.4.2.2.81 Mantenimiento de puestos de trabajo de un organismo

2.4.2.2.81.1 Insertar puesto de trabajo de un organismo

<p>void insertarPtoTrabOrganismo (TrPtoTrabOrganismo ptoTrabOrg, TpoString idPuestoTrab, TpoPK idOrganismo) throws TrException</p>

Relaciona un puesto de trabajo con un organismo Si todo es correcto devuelve los identificadores del puesto de trabajo y del organismo, si algo falla lanza una excepción.

Entradas

TrPtoTrabOrganismo Puesto de trabajo y organismo a relacionar.
ptoTrabOrg

Entrada / Salida

<i>TpoString</i> idPuestoTrab	Identificador del puesto de trabajo
<i>TpoPK</i> idOrganismo	Identificador del organismo

2.4.2.2.81.2 Eliminar puesto de trabajo de un organismo

```
int eliminarPtoTrabOrganismo( String idPuestoTrab,  
                             TpoPK idOrganismo ) throws TrException
```

Elimina la relación existente entre un puesto de trabajo y un organismo. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

<i>String</i> idPuestoTrab	Identificador del puesto de trabajo
<i>TpoPK</i> idOrganismo	Identificador del organismo

Salida

<i>int</i>	Número de filas eliminadas
------------	----------------------------

2.4.2.2.81.3 Obtener puesto de trabajo de un organismo

```
TrPtoTrabOrganismo[ ] obtenerPtoTrabOrganismo( String idPuestoTrab,  
                                                TpoPK idOrganismo,  
                                                ClausulaWhere where,  
                                                ClausulaOrderBy orderBy ) throws TrException
```

Permite obtener uno o varios puestos de trabajos relacionados a un organismo. Si se pasa sólo el parámetro *idPuestoTrab* se obtiene ese puesto de trabajo de todos los organismos, si se pasa sólo el parámetro *idOrganismo* se obtienen todos los puestos de trabajo de ese organismo y si se pasan ambos *null*, se devuelven todos.

Entradas

<i>String</i> idPuestoTrab	Identificador del puesto de trabajo
----------------------------	-------------------------------------

<i>TpoPK idOrganismo</i>	Identificador del organismo
<i>ClausulaWhere where</i>	Filtro a aplicar. Consultar campos posibles para TrPtoTrabOrganismo
<i>ClausulaOrderBy orderBy</i>	Ordenación a aplicar. Consultar campos posibles para TrPtoTrabOrganismo

Salida

<i>TrPtoTrabOrganismo[]</i>	Array de objetos TrPtoTrabOrganismo
------------------------------	-------------------------------------

2.4.2.2.82 Mantenimiento de sistemas

2.4.2.2.82.1 Insertar sistema

TpoPK insertarSistema(TrSistema sistema) throws TrException

Inserta un nuevo sistema. Si todo es correcto devuelve el identificador del nuevo sistema, si algo falla lanza una excepción.

Entradas

TrSistema sistema	Sistema a insertar
-------------------	--------------------

Salida

<i>TpoPK</i>	Identificador del nuevo sistema, cero si no se ha insertado
--------------	---

2.4.2.2.82.2 Modificar sistema

int modificarSistema(TrSistema sistema) throws TrException

Modifica un sistema existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrSistema sistema Sistema a modificar

Salida

int Número de filas modificadas

2.4.2.2.82.3 Eliminar sistema

int eliminarSistema(TpoPK idSistema) throws TrException

Elimina un sistema existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idSistema Identificador del sistema a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.82.4 Obtener sistema

**TrSistema [] obtenerSistema(TpoPK idSistema,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException**

Permite obtener uno o varios sistemas. Si se pasa el parámetro *idSistema* se devuelve los datos de ese sistema, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idSistema Identificador del sistema a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrSistema

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrSistema

Salida

TrSistema[] Array de objetos TrSistema

int obtenerSistemaCount(TpoPK idSistema,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve el número de sistemas

Entradas

TpoPK idSistema Identificador del sistema a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrSistema

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrSistema

Salida

int Número de registros

2.4.2.2.83 Mantenimiento de tipos de identificadores

2.4.2.2.83.1 Insertar tipo de identificador

String insertarTipIdentificador(TrTipIdentificador tipIdent) throws TrException

Inserta un nuevo tipo de identificador. Si todo es correcto devuelve el identificador del nuevo tipo de identificador, si algo falla lanza una excepción.

Entradas

TrTipIdentificador tipIdent Tipo de identificador a insertar.

Salida



String

Identificador del nueva tipo de identificador.

2.4.2.2.83.2 Modificar tipo de identificador

int modificarTipIdentificador(TrTipIdentificador tipIdent) throws TrException

Modifica un tipo de identificador existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipIdentificador tipIdent Tipo de identificador a modificar

Salida

int Número de filas modificadas

2.4.2.2.83.3 Eliminar tipo de identificador

int eliminarTipIdentificador(String idTipIdent) throws TrException

Elimina un tipo de identificador existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipIdent Identificador del tipo de identificador a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.83.4 Obtener tipo de identificador

**TrTipIdentificador [] obtenerTipIdentificador(String idTipIdent,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException**

Permite obtener uno o varios tipos de identificadores. Si se pasa el parámetro *idTipIdent* se devuelve los datos de ese tipo de identificador, si se pasa *null*, se devuelven todos.

Entradas

String idTipIdent Identificador del tipo de identificador a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoidentificador

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoidentificador

Salida

TrTipoidentificador[] Array de objetos TrTipoidentificador

2.4.2.2.84 Mantenimiento de tipos de organismos

2.4.2.2.84.1 Insertar tipo de organismo

TpoPK insertarTipoOrganismo (TrTipoOrganismo tipoOrg) throws TrException
--

Inserta un nuevo tipo de organismo. Si todo es correcto devuelve el identificador del nuevo tipo de organismo, si algo falla lanza una excepción.

Entradas

TrTipoOrganismo tipoOrg Tipo de organismo a insertar

Salida

TpoPK

Identificador del nuevo tipo de organismo, cero si no se ha insertado

2.4.2.2.84.2 Modificar tipo de organismo

int modificarTipoOrganismo(TrTipoOrganismo tipoOrg) throws TrException

Modifica un tipo de organismo existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoOrganismo tipoOrg

Tipo de organismo a modificar

Salida

int

Número de filas modificadas

2.4.2.2.84.3 Eliminar tipo de organismo

int eliminarTipoOrganismo(TpoPK idTipoOrg) throws TrException

Elimina un tipo de organismo existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoOrg

Identificador del tipo de organismo a eliminar

Salida

int

Número de filas eliminadas

2.4.2.2.84.4 Obtener tipo de organismo

TrTipoOrganismo [] obtenerTipoOrganismo(TpoPK idTipoOrg, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios tipos de organismos. Si se pasa el parámetro *idTipoOrg* se devuelve los datos de ese tipo de organismo, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoOrg

Identificador del tipo de organismo a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoOrganismo

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoOrganismo

Salida

TrTipoOrganismo[] Array de objetos TrTipoOrganismo

2.4.2.2.85 Mantenimiento de tipos de organizaciones

2.4.2.2.85.1 Insertar tipo de organización

String insertarTipoOrganizacion(TrTipoOrganizacion tipoOrgz) throws TrException

Inserta un nuevo tipo de organización. Si todo es correcto devuelve el identificador del nuevo tipo de organización, si algo falla lanza una excepción.

Entradas

TrTipoOrganizacion tipoOrgz Tipo de organización a insertar

Salida

String Identificador del nuevo tipo de organización.

2.4.2.2.85.2 Modificar tipo de organización

int modificarTipoOrganizacion(TrTipoOrganizacion tipoOrgz) throws TrException

Modifica un tipo de organización existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoOrganizacion tipoOrgz Tipo de organización a modificar

Salida

int Número de filas modificadas

2.4.2.2.85.3 Eliminar tipo de organización

int eliminarTipoOrganizacion(String idTipoOrgz) throws TrException

Elimina un tipo de organización existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoOrgz Identificador del tipo de organización a eliminar

Salida

int Número de filas eliminadas

2.4.2.2.85.4 Obtener tipo de organización

TrTipoOrganizacion [] obtenerTipoOrganizacion(String idTipoOrgz, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios tipos de organizaciones. Si se pasa el parámetro *idTipoOrgz* se devuelve los datos de ese tipo de organización, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idTipoOrgz Identificador del tipo de organización a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoOrganizacion

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoOrganizacion

Salida

TrTipoOrganizacion[] Array de objetos TrTipoOrganizacion

2.4.2.2.86 Mantenimiento de tipos de vías

2.4.2.2.86.1 Insertar tipo de vía

String insertarTipoVia (TrTipoVia tipoVia) throws TrException

Inserta un nuevo tipo de vía. Si todo es correcto devuelve el identificador del nuevo tipo de vía, si algo falla lanza una excepción.

Entradas

TrTipoVia tipoVia	Tipo de vía a insertar
-------------------	------------------------



Salida

String

Identificador del nuevo tipo de vía.

2.4.2.2.86.2 Modificar tipo de vía

int modificarTipoVia(TrTipoVia tipoVia) throws TrException

Modifica un tipo de vía existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrTipoVia tipoVia

Tipo de vía a modificar

Salida

int

Número de filas modificadas

2.4.2.2.86.3 Eliminar tipo de vía

int eliminarTipoVia(String idTipoVia) throws TrException

Elimina un tipo de vía existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

TpoPK idTipoVia

Identificador del tipo de vía a eliminar

Salida

int

Número de filas eliminadas

2.4.2.2.86.4 Obtener tipo de vía

**TrTipoVia [] obtenerTipoVia(String idTipoVia,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException**

Permite obtener uno o varios tipos de vías. Si se pasa el parámetro *idTipoVia* se devuelve los datos de ese tipo de vía, si se pasa *null*, se devuelven todas.

Entradas

TpoPK idTipoVia

Identificador del tipo de vía a obtener. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrTipoVia

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrTipoVia

Salida

TrTipoVia[] Array de objetos TrTipoVia

2.4.2.2.87 Mantenimiento de usuarios

2.4.2.2.87.1 Insertar usuario

String insertarUsuario(TrUsuario usuario) throws TrException

Inserta un nuevo usuario. Si todo es correcto devuelve el identificador del nuevo usuario, si algo falla lanza una excepción.

Entradas

TrUsuario usuario Usuario a insertar

Salida

String

Identificador del nuevo usuario.

2.4.2.2.87.2 Modificar usuario

```
int modificarUsuario( TrUsuario usuario ) throws TrException
```

Modifica un usuario existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

TrUsuario usuario

Usuario a modificar

Salida

int

Número de filas modificadas

2.4.2.2.87.3 Modificar clave usuario

```
int modificarUsuario( String codUsuario, String nuevaClave ) throws TrException
```

Modifica la clave de un usuario existente. Si todo es correcto devuelve el número de filas modificadas, si algo falla lanza una excepción.

Entradas

String codUsuario

Usuario a modificar

String nuevaClave

Nueva clave del usuario

Salida

int

Número de filas modificadas

2.4.2.2.87.4 Eliminar usuario

```
int eliminarUsuario( String idUsuario) throws TrException
```

Elimina un usuario existente. Si todo es correcto devuelve el número de filas eliminadas, si algo falla lanza una excepción.

Entradas

String idUsuario

Identificador del usuario a eliminar

Salida

int

Número de filas eliminadas

2.4.2.2.87.5 Obtener usuario

TrUsuario [] obtenerUsuario(String idUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy
Boolean sinPerfil) throws TrException

Permite obtener uno o varios usuarios. Si se pasa el parámetro *idUsuario* se devuelve los datos de ese usuario, si se pasa *null*, se devuelven todos. Permite obtener usuarios que no tengan perfiles asociados.

Entradas

<i>String</i> idUsuario	Identificador del usuario a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrUsuario
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrUsuario
<i>Booleab</i> sinPerfil	Filtro para obtener usuarios sin perfil asociado.

Salida

TrUsuario[] Array de objetos TrUsuario

int obtenerUsuarioCount(String idUsuario,
ClausulaWhere where,
ClausulaOrderBy orderBy) throws TrException

Devuelve el número de usuarios

Entradas

<i>String</i> idUsuario	Identificador del usuario a obtener. Si se pasa <i>null</i> se obtienen todos.
<i>ClausulaWhere</i> where	Filtro a aplicar. Consultar campos posibles para TrUsuario
<i>ClausulaOrderBy</i> orderBy	Ordenación a aplicar. Consultar campos posibles para TrUsuario

Salida

int Número de registros

2.4.2.2.88 Mantenimiento de días festivos

2.4.2.2.88.1 Insertar día festivo

TpoPK insertarDiaFestivo(TrDiaFestivo diaFestivo) throws TrException

Inserta un nuevo día festivo. Se no se indica código de provincia ni código de municipio se considera un festivo nacional, si sólo se indica el código de la provincia se considera festivo provincial y si se indican los códigos de provincia y municipio se considera festivo municipal o local.

Entradas

TrDiaFestivo diaFestivo

Día festivo a insertar

Salida

TpoPK Identificador del nuevo día festivo

2.4.2.2.88.2 Modificar día festivo

int modificarDiaFestivo(TrDiaFestivo diaFestivo) throws TrException

Modifica día festivo existente. Se no se indica código de provincia ni código de municipio se considera un festivo nacional, si sólo se indica el código de la provincia se considera festivo provincial y si se indican los códigos de provincia y municipio se considera festivo municipal o local.

Entradas

TrDiaFestivo diaFestivo Día festivo a modificar. Se debe indicar su REF DIA FESTIVO.

Salida

int Número de filas modificadas

2.4.2.2.88.3 Eliminar día festivo

int eliminarDiaFestivo(TpoPK idDiaFestivo) throws TrException

Elimina el día festivo indicado en el parámetro idDiaFestivo.

Entradas

TpoPK idDiaFestivo Identificador del día festivo

Salida

int Número de filas eliminadas

2.4.2.2.88.4 Obtener días festivos

TrDiaFestivo [] obtenerDiaFestivo(String idDiaFestivo, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Permite obtener uno o varios días festivos. Si se pasa el parámetro *idDiaFestivo* se devuelve los datos de ese día festivo, si se pasa *null*, se devuelven todos.

Entradas

TpoPK idDiaFestivo Identificador del día festivo. Si se pasa *null* se obtienen todos.

ClausulaWhere where Filtro a aplicar.
Consultar campos posibles para TrDiafestivo

ClausulaOrderBy orderBy Ordenación a aplicar.
Consultar campos posibles para TrDiafestivo

Salida

TrDiafestivo [] Array de objetos TrDiafestivo

2.4.2.2.88.5 Exportar calendario a XML

```
void exportarCalendarioXML( String codProvincia,
                           String provincia,
                           String codMunicipio,
                           String municipio,
                           boolean incluirNacionales,
                           boolean incluirProvinciales,
                           boolean incluirLocales,
                           OutputStream out) throws TrException
```

Exporta en un documento xml los festivos definidos para la provincia y/o municipio indicado

Entradas

- String* codProvincia Código de la provincia
- String* provincia Provincia
- String* codMunicipio Código del municipio
- String* municipio Municipio
- boolean* incluirNacionales Indica si se incluyen los festivos nacionales
- boolean* incluirProvinciales Indica si se incluyen los festivos provinciales
- boolean* incluirLocales Indica si se incluyen los festivos locales o municipales
- OutputStream* out Flujo de salida dónde se escribirá el xml

2.4.2.2.88.6 Importar calendario a XML

void importarCalendarioXML(InputStream in
boolean actualizar) throws TrException

Exporta en un documento xml los festivos definidos para la provincia y/o municipio indicado

Entradas

InputStream in

Flujo de entrada dónde se leerá el xml

boolean actualizar

Indica si se actualiza la descripción y/o periódico del festivo en caso de que ya existiese.

2.4.2.2.89 Otras APIs

2.4.2.2.89.1 Establecer el usuario

void establecerUsuarioSistema(String usuario) throws TrException

Permite establecer el usuario del TrAPIADM.

Entradas

<i>String usuario</i>	<i>Usuario</i>
-----------------------	----------------

Los usuarios deben tener el rol o perfil de usuario TR_R_USUARIO o el TR_R_ADMINISTRADOR.

2.4.2.2.89.2 Obtener usuario establecido

String obtenerUsuarioSistema(TpoString rol)

Permite obtener el usuario establecido en el TrAPIADM y su rol o perfil.

Entrada/Salida

<i>TpoString rol</i>	<i>Rol del usuario establecido en el api</i> <i>Parámetro de entrada/salida</i>
----------------------	--

Salida

<i>String usuario</i>	<i>Usuario establecido en el api</i>
-----------------------	--------------------------------------

2.4.2.2.89.3 Cierre de la sesión

void cerrarSesion()

Cierra la sesión actual de trabajo, haciendo por defecto un commit si el autoCommit está a true, si no se realiza un rollback.

void cerrarSesion(boolean commit)

Cierra la sesión actual de trabajo. Realiza un commit o un rollback según se indique en el parámetro *commit*.

Entrada

<i>boolean commit</i>	→ "true" – Se realiza un commit para guardar los cambios. → "false" – Se realiza un rollback para no guardar los cambios, desde el último commit.
-----------------------	--

2.4.2.2.89.4 AutoCommit

void setAutoCommit(boolean value)

Establece el valor del autoCommit.

Entrada

boolean value

→ **“true”** – Se realiza un commit cada vez que se ejecute un api.
 → **“false”** – No se realiza commit cuando se ejecute un api, el usuario debe hacer uso del método commit() para guardar los cambios.

boolean getAutoCommit()

Devuelve el estado actual del autoCommit.

Salida

boolean

→ **“true”** – Se realiza un commit cada vez que se ejecute un api.
 → **“false”** – No se realiza commit cuando se ejecute un api, el usuario debe hacer uso del método commit() para guardar los cambios.

2.4.2.2.89.5 Commit y Rollback

boolean commit()

Realiza un commit sobre la base de datos guardando los últimos cambios.

Salida

boolean

→ **“true”** – El commit se ha realizado correctamente
 → **“false”** – El commit no se ha realizado correctamente

boolean rollback()

Realiza un rollback sobre la base de datos deshaciendo los últimos cambios.

Salida

boolean

→ **“true”** – El rollback se ha realizado correctamente
 → **“false”** – El rollback no se ha realizado correctamente

2.4.2.2.89.6 Comprobación de conexión

boolean hayConexion()

Indica si existe o no una conexión válida a la base de datos.

Salida

boolean

→ **“true”** – Existe una conexión válida a la base de datos.
 → **“false”** – No existe conexión válida a la base de datos.

2.4.2.2.89.7 Establecer conexión fija

synchronized void establecerConexionFija(boolean fija)

Permite establecer la conexión como fija, es decir una vez que se crea no se cierra hasta que no se ejecute el método cerrarSesion. Este método sólo tiene efecto cuando la TrAPIADM se ha creado con un perfil de conexión que sea un datasource o un properties con pool.

Si se está usando un perfil de conexión que es un datasource o un properties con pool, y el conexionFija es false se devuelve la conexión al finalizar la ejecución de un método, en caso contrario no se devuelve.

Entrada

boolean fija

Indica si la conexión es fija para la TrAPIADM o se devuelve al pool de conexiones al finalizar la ejecución de un método.

→ **"true"** – La conexión es fija y no se devuelve al pool. Por defecto toma este valor.

→ **"false"** – Se devuelve la conexión al pool al terminar la ejecución de cada método, haciendo se commit al final de cada método.

2.4.2.2.89.80 Obtener estado conexión fija

synchronized boolean obtenerEstadoConexionFija()

Devuelve el valor actual del atributo conexionFija. El estado de la conexión fija sólo tiene efecto cuando la TrAPIADM se ha creado usando un perfil de conexión que es un datasource o un properties con pool. Si el valor devuelto es true indica que la conexión se fija y no se devuelve al pool hasta cerrar la sesión, en caso contrario si el valor es false indica que no es fija y que se devuelve la conexión al pool al finalizar la ejecución de cada método.

Salida

boolean

Valor del atributo conexionFija

→ **"true"** – La conexión es fija y no se devuelve al pool. Por defecto toma este valor.

→ **"false"** – Se devuelve la conexión al pool al terminar la ejecución de cada método, haciendo se commit al final de cada método.

2.4.2.2.90 APIs exclusivas para Expediente Electrónico

2.4.2.2.90.1 Obtener tipos de documentos ENI

TrTipoDocumentoEni[] obtenerTiposDocumentosENI (TpoPK idTipDocEni, ClausulaWhere where, ClausulaOrderBy orderBy) throws TrException

Obtiene un conjunto de tipos de documentos ENI.

Entrada/Salida

<i>TpoPK idTipDocEni</i>	<i>Identificador de tipo de documentos ENI</i>
<i>ClausulaWhere where</i>	<i>Cláusula where para la consulta</i>
<i>ClausulaOrderBy orderBy</i>	<i>Cláusula orderBy para la consulta</i>

Salida

<i>TrTipoDocumentoEni[]</i>	<i>Array de tipos de documentos ENI</i>
-----------------------------	---

2.4.2.2.90.2 Insertar tipo de documento ENI

TpoPK insertarTipoDocumentoENI (TrTipoDocumentoEni tipoDocEni) throws TrException

Inserta un tipo de documento ENI.

Entrada/Salida

<i>TrTipoDocumentoEni tipoDocEni</i>	<i>Identificador de tipo de documentos ENI</i>
--------------------------------------	--

Salida

<i>TpoPK</i>	<i>Identificador del tipo de documento ENI insertado</i>
--------------	--

2.4.2.2.90.3 Modificar tipo de documento ENI

int modificarTipoDocumentoENI (TrTipoDocumentoEni tipoDocEni) throws
TrException

Modifica un tipo de documento ENI.

Entrada/Salida

<i>TrTipoDocumentoEni</i>	<i>Tipo de documento ENI a modificar</i>
tipoDocEni	

Salida

<i>int</i>	<i>Número de filas modificadas</i>
------------	------------------------------------

2.4.2.2.90.4 Eliminar tipo de documento ENI

int eliminarTipoDocumentoENI (TpoPK idTipoDocEni) throws TrException

Elimina un tipo de documento ENI.

Entrada/Salida

<i>TpoPK idTipDocEni</i>	<i>Identificador de tipo de documento ENI</i>
--------------------------	---

Salida

<i>int</i>	<i>Número de filas eliminadas</i>
------------	-----------------------------------

2.4.3 Descripción del TrAPIUTL

2.4.3.1 Descripción de clases involucradas

2.4.3.1.1 Interfaz J-TrAPIUTL

La interfaz J-TrAPIUTL define todos los métodos para facilitar el intercambio de definiciones de procedimiento entre diferentes sistemas con TREW@. En la librería este acceso se realiza a través de la implementación de la interfaz **TrAPIUTL** que se describe en este apartado.

2.4.3.1.1.1 TrAPIUTL

trewa.bd.trapi.trapiutl.TrAPIUTL

Interfaz de acceso al motor de tramitación TREW@.

Para poder trabajar con la TrAPIUTL es preciso crear una instancia de la misma a través de la clase Factoría: *TrAPIUTLFactory*, descrita en el apartado siguiente.

Métodos para la creación de instancias del TrAPIUI

2.4.3.1.1.2 Descarga de procedimientos

→ `InputStream descargaDDP(TpoPK idDDP, TpoString error)`
deprecated

Descarga el XML asociado a una definición de procedimientos

<u>Entradas</u>	TpoPK idDDP TpoString error	<i>Identificador de la definición de procedimiento. Mensaje de error Entrada/Salida</i>
<u>Salidas</u>	InputStream	<i>Fichero XML con la información referente a la definición de procedimiento.</i>

→ `void descargaDDP(TpoPK idDDP, OutputStream out, TrAPIUTLIO archivos) throws TrException`

Descarga el XML asociado a una definición de procedimiento y la referencia a los archivos asociados

<u>Entradas</u>	TpoPK idDDP OutputStream out TrAPIUTLIO archivos	<i>Identificador del expediente. Flujo de salida del fichero XML del DDP Lista con los nombres de los archivos adjuntos que se tendrán que descargar. Ver API de descarga de archivos.</i>
------------------------	--	--

2.4.3.1.1.3 Descarga de expedientes

→ InputStream descargaExpediente(TpoPK idExpediente, TpoString error)
deprecated

Descarga el XML asociado a la evolución de un expediente

<u>Entradas</u>	TpoPK idExpediente TpoString error	<i>Identificador del expediente. Mensaje de error Entrada/Salida</i>
<u>Salidas</u>	InputStream	<i>Fichero XML con la información referente a la evolución de un expediente.</i>

→ void descargaExpediente(TpoPK idExpediente, OutputStream out, TrAPIUTLIO archivos) throws TrException

Descarga el XML asociado a la evolución de un expediente

<u>Entradas</u>	TpoPK idExpediente OutputStream out TrAPIUTLIO archivos	<i>Identificador del expediente. Flujo de salida del fichero XML del Expediente Lista con los nombres de los archivos adjuntos que se tendrán que descargar. Ver API de descarga de archivos.</i>
------------------------	---	---

2.4.3.1.1.4 Subida de definición de procedimientos

→ String subidaDDP(boolean actualizar, InputStream ficheroDDP, int size)
deprecated

Sube un XML que representa una definición de procedimiento.

Entradas	Boolean InputStream ficheroDDP Int size String	<i>Indica si se actualiza o no la información existente en Trew@ con la que viene en el XML.</i> → “ S ” – Se actualiza la información que exista con la que viene en el XML. → “ N ” – No se actualiza la información que exista con la que viene en el XML. <i>Archivo XML con la definición de procedimiento.</i> <i>Tamaño del archivo XML.</i> <i>Mensaje de error.</i>
Salidas		

→ TpoPK subidaDDP(boolean actualizar,InputStream ficheroDDP,TrAPIUTLIO flujos) throws TrException

Sube un XML que representa una definición de procedimiento.

Entradas	boolean InputStream ficheroDDP TrAPIUTLIO flujos	<i>Indica si se actualiza o no la información existente en Trew@ con la que viene en el XML.</i> → “ S ” – Se actualiza la información que exista con la que viene en el XML. → “ N ” – No se actualiza la información que exista con la que viene en el XML en este caso el procedimiento intentará ser borrado previo a su inserción. <i>Archivo XML con la definición de procedimiento.</i> <i>Flujos flujos de entrada correspondientes a los archivos.</i> <i>Ver API de subida de archivos.</i>
Salidas	TpoPK	<i>Identificador del nuevo procedimiento o del procedimiento actualizado en la base de datos.</i>

2.4.3.1.1.5 Descarga de entidades del sistema

→ void descargaDES(TpoPK idSistema, OutputStream out, TrAPIUTLIO archivos) throws TrException

Descarga el XML asociado a la evolución de un expediente

Entradas	TpoPK idSistema OutputStream out TrAPIUTLIO archivos	<i>Identificador del sistema</i> <i>Flujo de salida del fichero XML del DES</i> <i>Lista con los nombres de los archivos adjuntos que se tendrán que descargar.</i> <i>Ver API de descarga de archivos.</i>
-----------------	--	--

2.4.3.1.1.6 Obtención de archivos

→ void obtenerArchivo(TrAPIUTLArchivo archivo,OutputStream oArchivo) throws TrException;

Se obtiene el archivo de BBDD

<u>Entradas</u>	TrAPIUTLArchivo archivo OutputStream oArchivo	<i>Información del archivo Flujo de salida hacia el archivo</i>
------------------------	--	---

2.4.3.1.1.7Anexar archivos

→ void anexarArchivo(TrAPIUTLArchivo archivo,InputStream iArchivo) throws TrException

Inserta una archivo en la base de datos

<u>Entradas</u>	TrAPIUTLArchivo archivo InputStream iArchivo	<i>Identificador del sistema Flujo de entrada desde el que se lee el archivo</i>
------------------------	---	--

2.4.3.1.1.8Bloqueo y desbloqueo de procedimientos

→ void bloquearDefProcedimiento(TpoPK idDefProc) throws TrException

Bloquea el procedimiento indicado siempre que no esté bloqueado por otro usuario

<u>Entradas</u>	TpoPK idDefProc	<i>Identificador de la definición de procedimiento</i>
------------------------	-----------------	--

→ void desbloquearDefProcedimiento(TpoPK idDefProc) throws TrException

Desbloquea el procedimiento indicado si está bloqueado por el mismo usuario

<u>Entradas</u>	TpoPK idDefProc	<i>Identificador de la definición de procedimiento</i>
------------------------	-----------------	--

→ String procedimientoBloqueadoPor (TpoPK idDefProc, TpoString nombre) throws TrException

Indica qué usuario tiene bloqueado un procedimiento

<u>Entradas</u>	TpoPK idDefProc TpoString nombre	<i>Identificador de la definición de procedimiento Nombre y apellidos del usuario que tiene bloqueado el procedimiento (Salida)</i>
<u>Salida</u>	String	<i>Código de usuario que tiene bloqueado el procedimiento</i>

2.4.3.1.1.9Borrar procedimiento

→ void borrarProcedimiento(TpoPK idDefProc, String tipoBorrado) throws TrException

Borra el procedimiento indicado. Mediante el parámetro tipoBorrado indicamos hasta qué nivel queremos borrar, invalidando las entidades que no sea posible borrar.

<u>Entradas</u>	TpoPK idDefProc String tipoBorrado	<i>Identificador del procedimiento</i>
		→ “B” – Básico. Se realiza un borrado (o invalidación) básico del procedimiento, es decir sin borrar fases, tipos de documentos, tareas, etc.
		→ “A” – Avanzado. Se realiza un borrado (o invalidación) avanzado, es decir borra fases, metafases, transiciones, tipos documentos, tareas, etc.
		→ “I” – Intermedio. Se realiza un borrado (o invalidación) intermedio, similar al avanzado, sin eliminar versiones del procedimiento ni relaciones.

2.4.3.1.1.10Otros métodos

→ boolean hayConexion()

Indica si existe o no una conexión válida a la base de datos

<u>Salidas</u>	boolean	→ “true” – Existe una conexión válida a la base de datos. → “false” – No existe conexión válida a la base de datos.
-----------------------	---------	--

→ void cerrarSesion()

Cierra la sesión actual de trabajo haciendo un commit por defecto si está activado el autoCommit

→ void cerrarSesion(boolean commit)

Cierra la sesión actual de trabajo indicándose si se realiza commit o no

<u>Entradas</u>	boolean	<i>Indica si se hace o no el autoCommit</i>
		→ “S” – Hacer commit
		→ “N” – No hacer commit

→ void setAutoCommit(boolean value)

Permite establece el valor del autoCommit

Entradas boolean

Indica si se activa o no el autoCommit

→ “**S**” – Autocommit activado

→ “**N**” – AutoCommit no activado. Se precisará de control de commit explícito por parte de la aplicación cliente..

→ void getAutoCommit()

Devuelve el valor del autoCommit

Salida boolean

Valor de autoCommit establecido actualmente

→ void commit()

Realiza un commit en la base de datos, salvando los cambios

→ boolean rollback()

Realiza un rollback en la base de datos lo que deshace los últimos cambios

Salida boolean

Devuelve true si el rollback se hizo con éxito y false en caso contrario

2.4.3.1.2 Clase Factoría

La clase Factoría se encarga de crear una instancia correcta de la J-TrAPIUTL de forma que los métodos de esta última aparezcan disponibles para las aplicaciones.

2.4.3.1.2.1 TrAPIUTLFactory

trewa.bd.trapi.trapiutl.TrAPIUTLFactory

Clase que aporta funcionalidad para la correcta creación de una instancia de la TrAPIUTL.

Métodos para la creación de instancias del TrAPIUTL

→ TrAPIUTL crearAPIUTL()

Devuelve una instancia del TrAPIUTL. La información para el establecimiento de la conexión la obtiene a través del perfil "default".

Salidas TrAPIUTL Instancia del TrAPIUTL o "null" si no se pudo crear.

→ TrAPIUTL crearAPIUTL(String usuario, String clave)

Devuelve una instancia del TrAPIUTL. La información para el establecimiento de la conexión la obtiene a través del perfil "default" pero en este caso hace uso del usuario y password que recibe como parámetros.

Entradas *String usuario* Usuario
String clave Clave de usuario

Salidas TrAPIUTL Instancia del TrAPIUTL o "null" si no se pudo crear.

→ TrAPIUTL crearAPIUTL(String perfil)

Devuelve una instancia del TrAPIUTL. La información para el establecimiento de la conexión la obtiene a través del perfil especificado en la llamada como parámetro.

Entradas *String perfil* *Nombre del perfil para el establecimiento de la conexión.*

Salidas TrAPIUTL Instancia del TrAPIUTL o "null" si no se pudo crear.

→ TrAPIUTL crearAPIUTL(String perfil)

Devuelve una instancia del TrAPIUTL. La información para el establecimiento de la conexión la obtiene a través del perfil especificado en la llamada como parámetro.

Entradas *String perfil* *Nombre del perfil para el establecimiento de la conexión.*
String usuario *Usuario*
String clave *Clave de usuario*

Salidas TrAPIUTL *Instancia del TrAPIUTL o "null" si no se pudo crear.*

Ejemplo de uso

En el siguiente ejemplo se muestra cómo obtener una instancia del TrAPIUTL a través de la clase TrAPIUTLFactory:

```

TrAPIUTL apiUTL = null;
InputStream is = null;
TpoPK idExpediente = null;
TpoString strErr = null;

apiUTL = TrAPIUTLFactory.crearAPIUI( "perfilTrewa" );
if (apiUTL!= null ){
    // Ya se puede acceder a los métodos del TrAPIUTL
    idExpediente = new TpoPK( BigDecimal.valueOf(2) );
    strErr = new TpoString("");
    try{
        is = apiUTL.descargaExpediente( idExpediente, strErr );
        ...
    }
    catch( TrException ex )
{
    ex.printStackTrace();
}
    catch( Exception ex )
{
    ex.printStackTrace();
}
}
}

```

2.4.3.1.3 Clases de entidad

Estas clases se usan para el intercambio de información entre las aplicaciones cliente y la interfaz TrAPIUTL.

2.4.3.1.3.1 TrAPIUTLArchivo

trewa.bd.trapi.trapiutl.TrAPIUTLArchivo

Clase que representa a un archivo para insertar u obtener de un sistema Trew@.

Atributos accesibles mediante métodos get/set

<u>Atributo</u>	<u>Tipo</u>	<u>Descripción</u>
→ id	TpoPK	Identificador del archivo.
→ nombre	String	Nombre del archivo.
→ tipo	String	Tipo de archivo
→ mime	String	Tipo de mime del archivo

2.4.3.1.3.2 TrAPIUTLIO

trewa.bd.trapi.trapiutl.TrAPIUTLIO

Clase que representa a una conjunto de archivos a descargar del sistema Trew@.

Métodos públicos

→ TrAPIUTLArchivo[] getArchivos()
Método que devuelve el conjunto de archivos.

2.4.3.1.4 Servlets de subida y descarga de XML (DDP, Expedientes y Entidades del sistema)

Para la subida y descarga de archivos XML, junto con la TrAPIUTL, se aportan tres servlets que interactúan con la misma.

2.4.3.1.5 Servlet DescargaXML

El servlet DescargaXML obtiene de TREW@ un fichero XML que puede ser una Definición de Procedimiento, un Expediente o una definición de componentes de un sistema según se le indique en la petición.

Los parámetros del servlet se indican en la siguiente tabla:

Parámetro	Ámbito	Ob.	Descripción
tipo	Request	Sí	Indica el tipo de XML que se desea obtener de entre los siguientes: → “ddp” -- Definición de procedimiento. → “expediente” -- Expediente. → “des” -- Definición de entidades del sistema (en este caso devuelve un zip si existen archivos asociados)
id	Request	Sí	Identificador clave de la Definición de procedimiento, del expediente o del sistema, según el tipo de archivo que se desee obtener.
perfil	Request	No	Perfil con el que se intentará realizar la conexión. @deprectated
usuario	Session	No	Usuario con el que se realiza la conexión. Si no se ha indicado trUsuarioConexion ya que éste último tiene prioridad. @deprecated
clave	Session	No	Password de usuario. Si no se ha indicado trClaveConexion ya que éste último tiene prioridad. @deprecated
usuarioAPI	Request	No	Usuario a establecer en la interfaz TrAPIUTL. Si no se ha indicado trUsuarioEstablecer ya que éste último tiene prioridad. @deprecated
cerrarSesion	Request	No	Indica si se quiere cerrar la sesión de la interfaz TrAPIUTL una vez terminada la ejecución del servlet si se recibe el TrAPIUTL en sesión.
apiUTL	Session	Sí/No	Interfaz TrAPIUTL. Opcional si se indican los siguientes atributos en sesión..

En caso de que no se encuentre el apiUTL en sesión, se pueden indicar los siguientes parámetros para crear la instancia del TrAPIUTL.

trPerfilConexion	Session	Sí/No	Indica el perfil con el que se creará el TrAPIUTL. Obligatorio siempre que no se encuentre el apiUI en sesión.
trUsuarioConexion	Session	No	Indica el usuario que se utilizará para crear la conexión con Trew@. Obligatorio si el trPerfilConexion indicado no está configurado con usuario y clave.
trClaveConexion	Session	No	Indica la clave del usuario que se utilizará para crear la conexión con Trew@. Obligatorio si el trPerfilConexion indicado no está configurado con usuario y clave.
trUsuarioEstablecer	Session	No	Indica el usuario que se establecerá en el TrAPIUTL una vez creada. Si no se indica se queda establecido el usuario de la conexión.

El servlet DescargaXML necesita una conexión con la base de datos para obtener la información necesaria. Para ello se hace uso de la interfaz TrAPIUTL. Existen varias formas de establecer dicha conexión:

1. Crear un objeto de la clase TrAPIUTL con la conexión a la base de datos establecida y poner dicho objeto en la sesión.

- Indicar mediante sesión el atributo trPerfilConexion con un perfil de conexión configurado con usuario y clave, y el atributo trUsuarioEstablecer.
- Indicar mediante sesión los atributos trPerfilConexion, trUsuarioConexion, trClaveConexion y trUsuarioEstablecer para que el servlet cree la instancia del TrAPIUTL.

Ejemplo de llamada al servlet DescargaXML

→ Llamada desde una URL

Descarga de procedimiento

<http://servidorWeb/aplicacionJSP/trewa/DescargaXML?tipo=ddp&id=406>

Descarga de entidades del sistema

<http://servidorWeb/aplicacionJSP/trewa/DescargaXML?tipo=des&id=5>

→ Llamada desde un javascript

`location.href='trewa/DescargaXML?tipo=des&id=2'`

2.4.3.1.6 Servlet SubidaXML

El servlet SubidaXML inserta en TREW@ la información obtenida de un archivo XML que consistirá en la Definición de un Procedimiento.

Los parámetros del servlet se indican en la siguiente tabla:

Parámetro	Ámbito	Ob.	Descripción
act	Request	No	Indica si se actualiza la información existente para la Definición de procedimiento. Puede ser "true" o "false". El valor por defecto es "false".

perfil	Request	No	Perfil con el que se intentará realizar la conexión. @deprectated
usuario	Session	No	Usuario con el que se realiza la conexión. Si no se ha indicado trUsuarioConexion ya que éste último tiene prioridad. @deprecated
clave	Session	No	Password de usuario. Si no se ha indicado trClaveConexion ya que éste último tiene prioridad. @deprecated
apiUTL	Session	Sí/No	Interfaz TrAPIUTL. Opcional si se indican los siguientes atributos en sesión..

En caso de que no se encuentre el apiUTL en sesión, se pueden indicar los siguientes parámetros para crear la instancia del TrAPIUTL.

trPerfilConexion	Session	Sí/No	Indica el perfil con el que se creará el TrAPIUTL. Obligatorio siempre que no se encuentre el apiUI en sesión.
trUsuarioConexion	Session	No	Indica el usuario que se utilizará para crear la conexión con Trew@. Obligatorio si el trPerfilConexion indicado no está configurado con usuario y clave.
trClaveConexion	Session	No	Indica la clave del usuario que se utilizará para crear la conexión con Trew@. Obligatorio si el trPerfilConexion indicado no está configurado con usuario y clave.
trUsuarioEstablecer	Session	No	Indica el usuario que se establecerá en el TrAPIUTL una vez creada. Si no se indica se queda establecido el usuario de la conexión.

2.4.3.1.7 Servlet SubidaXML en formato ZIP

El servlet SubidaXML inserta en Trew@ la información obtenida de un archivo ZIP que contiene la Definición de un Procedimiento junto con archivos adicionales asociados a la definición.

Los parámetros del servlet se indican en la siguiente tabla:

Parámetro	Ámbito	Ob.	Descripción
act	Request	No	Indica si se actualiza la información existente para la Definición de procedimiento. Puede ser "true" o "false". El valor por defecto es "false".
apiUTL	Session	Sí/No	Interfaz TrAPIUTL. Opcional si se indican los siguientes atributos en

sesión..

En caso de que no se encuentre el apiUTL en sesión, se pueden indicar los siguientes parámetros para crear la instancia del TrAPIUTL.

trPerfilConexion	Session	Sí/No	Indica el perfil con el que se creará el TrAPIUTL. Obligatorio siempre que no se encuentre el apiUI en sesión.
trUsuarioConexion	Session	No	Indica el usuario que se utilizará para crear la conexión con Trew@. Obligatorio si el trPerfilConexion indicado no está configurado con usuario y clave.
trClaveConexion	Session	No	Indica la clave del usuario que se utilizará para crear la conexión con Trew@. Obligatorio si el trPerfilConexion indicado no está configurado con usuario y clave.
trUsuarioEstablecer	Session	No	Indica el usuario que se establecerá en el TrAPIUTL una vez creada. Si no se indica se queda establecido el usuario de la conexión.

→ synchronized void establecerConexionFija(boolean fija)

Permite establecer la conexión como fija, es decir una vez que se crea no se cierra hasta que no se ejecute el método cerrarSesion. Este método sólo tiene efecto cuando la TrAPIUTL se ha creado con un perfil de conexión que sea un datasource o un properties con pool.

Si se está usando un perfil de conexión que es un datasource o un properties con pool, y el conexionFija es false se devuelve la conexión al finalizar la ejecución de un método, en caso contrario no se devuelve.

Entrada	boolean fija	Indica si la conexión es fija para la TrAPIUTL o se devuelve al pool de conexiones al finalizar la ejecución de un método.
		→ "true" – La conexión es fija y no se devuelve al pool. Por defecto toma este valor.
		→ "false" – Se devuelve la conexión al pool al terminar la ejecución de cada método, haciendo se commit al final de cada método.

→ synchronized boolean obtenerEstadoConexionFija()

Devuelve el valor actual del atributo conexionFija. El estado de la conexión fija sólo tiene efecto cuando la TrAPIUTL se ha creado usando un perfil de conexión que es un datasource o un properties con pool. Si el valor devuelto es true indica que la conexión se fija y no se devuelve al pool hasta cerrar la sesión, en caso contrario si el valor es false indica que no es fija y que se devuelve la conexión al pool al finalizar la ejecución de cada método.

Salida boolean

Indica si la conexión es fija para la TrAPIUTL o se devuelve al pool de conexiones al finalizar la ejecución de un método.

→ **“true”** – La conexión es fija y no se devuelve al pool. Por defecto toma este valor.

→ **“false”** – Se devuelve la conexión al pool al terminar la ejecución de cada método, haciendo se commit al final de cada método.

→ void establecerUsuarioSistema(String usuario) throws TrException
Establece un usuario no oracle

Entradas String usuario *Nombre del usuario*

→ TrSistema[] obtenerSistemas() throws TrException
Devuelve los sistemas disponibles en Trew@

Salida TrSistema[] *Array de objetos TrSistema*

2.4.3.1.8 Servlet *JustificanteBorrado*

El servlet *JustificanteBorrado* inserta en TREW@ la información perteneciente al justificante para realizar el borrado lógico de TrewaAdm .

Los parámetros del servlet se indican en la siguiente tabla:

Parámetro	Ámbito	Ob.	Descripción
<i>file</i>	Request	Sí	Contiene el justificante de borrado.
<i>naos</i>	Request	No	Número del NAOS donde se solicita el borrado
<i>responsable</i>	Request	Sí	Persona responsable de realizar el borrado.
<i>observ</i>	Request	No	Observaciones a incluir en el borrado lógico
<i>adicional</i>	Request	No	Fichero adicional al justificante de borrado
<i>tipoBorrado</i>	Request	Sí	Variable que controla si se trata de un borrado de documento o expediente.
<i>idDocExp</i>	Request	No	Indica clave ajena al identificador del documento.
<i>idExp</i>	Request	No	Indica clave ajena al identificador del expediente.
<i>defProc</i>	Request	Sí	Indica el procedimiento al que pertenece el expediente/documento a borrar.
<i>apiADM</i>	Session	Sí/No	Interfaz TrAPIADM. Opcional si se indican los siguientes atributos en sesión.
<i>apiUI</i>	Session	Sí/No	Interfaz TrAPIUI. Opcional si se indican los siguientes atributos en sesión.
<i>idStma</i>	Session	Sí	Código del sistema al que pertenece el documento/expediente.

2.4.4 Despliegue y configuración de las J-TrAPIs

2.4.4.1 Integración con Maven

Se puede indicar la dependencia de la librería JTrAPI en proyectos Maven incluyendo la siguiente dependencia en el fichero pom.xml:

```
<groupId>es.juntadeandalucia.trewa</groupId>  
<artifactId>trewa-api</artifactId>  
<version>2.6.10.1</version>
```

Esta librería se encuentra desplegada en el repositorio Maven de la Consejería de Hacienda y Administración pública, accesible desde la siguiente dirección <https://ws074.juntadeandalucia.es/archiva/>.

2.4.4.2 Despliegue de las J-TrAPIs

Para que las J-TrAPIs puedan estar accesibles desde diferentes aplicaciones, deben estar desplegadas en el mismo servidor de aplicaciones donde dichas aplicaciones residan. El despliegue de las mismas variará en función del servidor de aplicaciones donde se vaya a realizar y en todo caso también se tendrá que realizar una configuración para habilitar el acceso a los servlets.

2.4.4.3 Configuración de servlets

Para que una aplicación pueda acceder a los servlets de subida y descarga de documentos (TrAPIUI), para los de subida y descarga de XML (TrAPIUTL) y para los de subida de documento WebOffice y descarga de XML WebOffice, debe existir un paso previo de configuración a nivel del servidor de aplicaciones. Básicamente la configuración consistirá en modificar el archivo “web.xml” del servidor para registrar la información de los servlets.

Se editará el archivo web.xml y en la primera parte “Built In Servlet Definitions” definiremos los servlets de la siguiente forma:



```
<servlet>
  <servlet-name>TrAPIUIDescarga</servlet-name>
  <servlet-class>trewa.bd.trapi.trapiui.servlet.DescargaDocumento</servlet-class>
</servlet>

<servlet>
  <servlet-name>TrAPIUISubida</servlet-name>
  <servlet-class>trewa.bd.trapi.trapiui.servlet.SubidaDocumento</servlet-class>
</servlet>
<servlet>
  <servlet-name>TrAPIUTLDescarga</servlet-name>
  <servlet-class>trewa.bd.trapi.trapiutl.servlet.DescargaXML</servlet-class>
</servlet>

<servlet>
  <servlet-name>TrAPIUTLSubida</servlet-name>
  <servlet-class>trewa.bd.trapi.trapiutl.servlet.SubidaXML</servlet-class>
</servlet>

<servlet>
  <servlet-name>TrAPIUTLSubidaZip</servlet-name>
  <servlet-class>trewa.bd.trapi.trapiutl.servlet.SubidaXMLzip</servlet-class>
</servlet>

<servlet>
  <servlet-name>JustificanteBorrado</servlet-name>
  <servlet-class>trewaAdm.servlets.JustificanteBorrado</servlet-class>
</servlet>

<!-- Servlets para WebOffice -->

<servlet>
  <servlet-name>TrAPIUISubidaOO</servlet-name>
  <servlet-class>trewa.bd.trapi.trapiui.servlet.OOSubidaDocumento</servlet-class>
</servlet>

<servlet>
  <servlet-name>TrAPIUIDescargaXMLOO</servlet-name>
  <servlet-class>trewa.bd.trapi.trapiui.servlet.OODescargaXML</servlet-class>
</servlet>
```



Además se debe añadir información referente a desde dónde se puede acceder a los servlets. Esta información se denomina “Servlet Mappings”, y para configurarla se incluirán las siguientes líneas en el apartado “Built In Servlet Mappings”:

```
<servlet-mapping>
  <servlet-name>TrAPIUIDescarga </servlet-name>
  <url-pattern>/trewa/DescargaDoc</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>TrAPIUISubida </servlet-name>
  <url-pattern>/trewa/SubidaDoc</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>TrAPIUTLDescarga </servlet-name>
  <url-pattern>/trewa/DescargaXML</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>TrAPIUTLSubida </servlet-name>
  <url-pattern>/trewa/SubidaXML</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>TrAPIUTLSubidaZip</servlet-name>
  <url-pattern>/trewa/SubidaXMLzip</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>TrAPIUISubidaOO</servlet-name>
  <url-pattern>/trewa/OOSubidaDoc/*</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>TrAPIUIDescargaXMLOO</servlet-name>
  <url-pattern>/trewa/OODescargaXML/*</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>JustificanteBorrado</servlet-name>
  <url-pattern>/trewaAdm/JustificanteBorrado</url-pattern>
</servlet-mapping>
```

La ubicación por defecto del archivo “**web.xml**” variará en función del servidor de aplicaciones:

2.4.4.4 Configuración de perfiles de conexión

Un perfil de conexión para las J-TrAPIs no es más que una configuración de la conexión a una base de datos.

En la actualidad las J-TrAPIs permite una sola configuración de los posibles perfiles de conexión:

- Configuración básica con archivo de properties

A nivel de aplicación el uso de un perfil u otro va a ser totalmente transparente y no se va a a conocer exactamente donde se establece la conexión. Lo único que van a necesitar las aplicaciones será el **nombre del perfil** que deberá ser aportado por el administrador del servidor de aplicaciones donde se desplieguen las J-TrAPIs.

2.4.4.5 Configuración con archivo de properties

En este caso el nombre del perfil va a coincidir con el nombre del archivo de properties donde se realice la configuración. Así por ejemplo si el administrador crea un archivo **trOracle.properties**, el nombre del perfil será **trOracle**.

Posibles ubicaciones para el archivo de perfiles

Los perfiles deben ubicarse en el paquete **trewa.conf.perfiles** para que las J-TrAPIs los puedan localizar. En esa carpeta debe existir configurado por lo menos un perfil **default** (default.properties) que es el que intentarán usar las J-TrAPIs cuando no se le indique perfil alguno.

En todo caso será conveniente analizar varias posibles situaciones para asegurarse de que el perfil sea detectado por las TrAPIs. En la tabla adjunta se muestra una comparativa de la ubicación del archivo trewa.jar y del archivo de configuración indicándose si la localización se llevará a cabo o no:

Ubicación del archivo trewa.jar en el servidor de aplicaciones	Ruta default.properties en el servidor de aplicaciones	Observaciones
TOMCAT		
common/lib	common/classes/trewa/conf/perfiles	El perfil es localizado por las TrAPIs
common/lib	webapps/<APLICACIÓN>/classes/trewa/conf/perfiles	El perfil no es localizado por las TrAPIs
webapps/<APLICACIÓN>/lib	common/classes/trewa/conf/perfiles	El perfil es localizado por las TrAPIs
webapps/<APLICACIÓN>/lib	webapps/<APLICACIÓN>/classes/trewa/conf/perfiles	El perfil es localizado por las TrAPIs
JBOSS		
server/default/lib	deploy/<APLICACIÓN.war>	El perfil no es localizado por las TrAPIs
deploy/<APLICACIÓN.war>	deploy/<APLICACIÓN.war>	El perfil es localizado por las TrAPIs

2.4.4.6 Formato del fichero de properties

La configuración del archivo de properties se realiza editando el archivo y publicando los siguientes campos:

Campo	Ob.	Descripción
tipo	Sí	Tipo de base de datos a la que se va a acceder. Por defecto ORACLE
nombreMaquina	Sí	Máquina
puerto	Sí	Puerto de acceso
nombreBD	Sí	Nombre de la base de datos
nombreUsuario	No	Nombre de usuario
claveUsuario	No	Clave de acceso
usaPool	No	Indica si se quiere hacer uso del pool interno que ofrecen las Jtrapis. Posibles valores: true ó false
maxConexionesActivas	No	Máximo de conexiones abiertas posibles
maxEsperaConexion	No	Tiempo máximo de espera (en milisegundos) para obtener una conexión del pool.
maxConexionesOciosas	No	Máximo de conexiones ociosas.

Un ejemplo de archivo de properties sería:

```
#Archivo de propiedades
tipo=ORACLE
#Datos para realizar la conexion
nombreMaquina=desa.ja.es
puerto=1521
nombreBD=bdOracle
nombreUsuario=
claveUsuario=

#Indica si se quiere crear la conexión usando pool de conexiones
usaPool=false

#Máximo de conexiones abiertas
maxConexionesActivas=4

#Máximo de tiempo de espera por conexión
maxEsperaConexion=5000

#Máximo de conexiones ociosas
maxConexionesOciosas=2
```

	Junta de Andalucía Agencia Digital de Andalucía	Trew@ Guía de referencia de JTrApi
---	---	---------------------------------------

Si no se especifica usuario, este será exigido a través de las J-TrAPIs para establecer la conexión. En este caso se dejaría el campo en el archivo pero sin valor asignado.

A partir de la versión 1.2.0 de Trew@ se permite indicar los valores de nombreMaquina, puerto y nombreBD en uno solo, formando así la url de conexión JDBC. Para ello basta con indicar la url de conexión JDBC en el atributo nombreMaquina. De este modo se permiten realizar conexiones remotas tan solo indicando la url de conexión JDBC.

Por ejemplo:

```
#Archivo de propiedades
tipo=ORACLE
#Datos para realizar la conexion
nombreMaquina=jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=
(PROTOCOL=TCP)(Host=nombreMaquina)(Port=puerto))(ADDRESS=(PROTOCOL = TCP)
(Host=nombreMaquina)(Port = puerto)))(CONNECT_DATA= (SID=nombreBD))
(SOURCE_ROUTE=YES))
puerto=
nombreBD=
nombreUsuario=
claveUsuario=
```

A partir de la versión 1.4.1 de Trew@ se permite indicar en el properties si se hace uso del pool de conexiones que ofrecen las JtrApis, para ello se definen los atributos usaPool, maxConexionesActivas, maxEsperaConexion y maxConexionesOciosas en el fichero properties.

2.4.4.7 Configuración generación documentos OpenOffice

Para la generación de documentos del tipo OpenOffice bean mediante las apis de Trew@ se debe configurar la constante del sistema "COMP_GEN_OO" cuyo valor debe ser el nombre de un componente en el cuál se pueden configurar los siguientes datos de componente:

Dato componente	Ob / Op	Descripción
OO_MARCA_VARIABLE	Op	Indica la marca para reconocer las variables. Por defecto su valor será "\$\$". Por ejemplo \$\$VARIABLE\$\$.
OO_PREFIJO_TABLA	Op	Indica el prefijo de las variables de tipo tabla. Por defecto su valor será "TAB_". Por ejemplo \$\$TAB_VARIABLE\$\$\$. Éstas variables devolverán un texto formateado con los separadores de fila y columna correspondientes.
OO_PREFIJO_IMAGEN	Op	Indica el prefijo de las variables de tipo imagen. Por defecto su valor será "IMG_". Por ejemplo \$\$IMG_VARIABLE\$\$\$. Éstas variables devolverán una ruta de una imagen.(http://... ó file:///...).
OO_SEPARADOR_FILA	Op	Indica el carácter separador de filas para variables de tipo tabla. Por defecto su valor será ",".
OO_SEPARADOR_COLUMNA	Op	Indica el carácter separador de columnas para variables de tipo tabla. Por defecto su valor será ",".
OO_CONVERT_PDF_PUERTO	Op	Indica el puerto del servidor de OpenOffice para la generación a PDF. Por defecto su valor será "8100".
OO_CONVERT_PDF_SERVIDOR	Op	Indica el servidor en el cuál está desplegado el servicio de OpenOffice para la generación a PDF. Por defecto su valor será "localhost".

Se recomienda utilizar el mismo componente definido para WebOffice ya que comparten éstos datos de componente excepto OO_CONVERT_PDF_PUERTO y OO_CONVERT_PDF_SERVIDOR que sólo se usan para la conversión a PDF desde el api generarDocumentoPDF. De esta forma si se edita un documento con WebOffice o se modifica mediante las JTrAPIs el comportamiento será el mismo ya que reemplazará las variables con la misma marca de variables, los mismos prefijos y separadores.



2.4.4.8 Trazabilidad con log4java

Para habilitar la trazabilidad de log4java en trewa, bastará con modificar un archivo:

1.- El archivo *log4j.properties*

A través de este archivo se podrá configurar el nivel de trazabilidad que se desea habilitar sobre trewa como la ruta donde se despliega el fichero log.

Para establecer la ruta del fichero log se debe modificar la etiqueta **filename** indicando la ruta en la que se desea crear el fichero.

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration>
  <Appenders>
    <File name="TREWA" fileName="<web_app>/logs/trewa.log">
      <PatternLayout pattern="%d{[yyyy-MM-dd HH:mm:ss,SSS]} %5p: %m%n"/>
    </File>
    <Console name="CONSOLE" target="SYSTEM_OUT">
      <PatternLayout pattern="%5p: %m%n"/>
    </Console>
    <Async name="ASYNC">
      <AppenderRef ref="TREWA" level="info"/>
      <AppenderRef ref="CONSOLE" level="info"/>
    </Async>
  </Appenders>
  <Loggers>
    <Root level="debug">
      <AppenderRef ref="ASYNC"/>
    </Root>
  </Loggers>
</Configuration>
```

Para cambiar la configuración de los niveles de trazabilidad sobre Trew@, se modificará la entrada *log4j.logger.trewa* estableciendo algún nivel de los posibles que se describen a continuación:

DEBUG: Se utiliza para escribir mensajes de depuración, este log no debe estar activado cuando la aplicación se encuentre en producción.

INFO: Se utiliza para mensajes similares al modo "verbose" en otras aplicaciones.

WARN: Se utiliza para mensajes de alerta sobre eventos que se desea mantener constancia, pero que no afectan el correcto funcionamiento del programa.

ERROR: Se utiliza en mensajes de error de la aplicación que se desea guardar, estos eventos afectan al programa pero lo dejan seguir funcionando.

FATAL: Se utiliza para mensajes críticos del sistema, generalmente luego de guardar el mensaje el programa abortará.

Adicionalmente a estos niveles de log, existen 2 niveles extras que solo se utilizan en el archivo de configuración, estos son:

ALL: este es el nivel más bajo posible, habilita todos los logs.

OFF: este es el nivel más alto posible, deshabilita todos los logs.

log4j.logger.trewa=OFF, FILE_LOG_TREWA_APPENDER

2.4.4.9 Configuración para WebOffice

Para el correcto funcionamiento de WebOffice (**versiones anteriores a v3.0.0**) se precisa el establecimiento de un perfil para el mismo. Dicho perfil se especificará a través del archivo de **weboffice.properties** que se ubicará en el paquete **trewa.conf** de la aplicación cliente con el siguiente formato:

perfil=default

Campo	Ob.	Descripción
perfil	No	Perfil de acceso para WebOffice

2.4.4.10 Despliegue de los Servicios Web de Trew@

Para el correcto funcionamiento de los servicios web de Trew@ se precisa realizar el despliegue de una aplicación con las siguientes características:

1.- Definición de los perfiles de acceso. Para esto se creará el archivo **trewaws.properties** en el paquete **trewa.conf**, donde se especificarán los perfiles a usar por los diferentes servicios. El archivo *trewaws.properties* tiene el siguiente formato:

```

perfil=java:comp/env/jdbc/trewaws
perfil_trewa=java:comp/env/jdbc/trewaws1
perfil_modelo=java:comp/env/jdbc/trewaws2
perfil_warda=java:comp/env/jdbc/trewaws3

```

Campo	Ob.	Descripción
perfil	No	Perfil por defecto para todos los servicios
perfil_trewa	No	Perfil por defecto para el servicio de trewa. Si no se especifica esta entrada se tomará la que se indique en la entrada <i>perfil</i> .
perfil_modelo	No	Perfil por defecto para el servicio de actualización de codificadoras. Si no se especifica esta entrada se tomará la que se indique en la entrada <i>perfil</i> .
perfil_warda	No	Perfil por defecto para el servicio de w@rda. Si no se especifica esta entrada se tomará la que se indique en la entrada <i>perfil</i> .

2.- Inclusión en la carpeta **public_html\WEB-INF** de la aplicación los archivos **web.xml** y **server-config.wsdd**. Ambos archivos se aportan junto con la plantilla de ejemplo para montar el servicio web.

3.- Inclusión de las librerías necesarias.

2.4.4.11 Condiciones, acciones, variables y selectores de plantillas Java

Una de las características que aporta la interfaz de acceso a Trew@ es la posibilidad de ejecutar clases java desplegadas en el servidor de aplicaciones. Para poder habilitar esta característica se deben tener en cuenta los siguientes puntos:

1.- Se debe definir alguna condición/acción/ variable y/o selector de plantilla en un procedimiento con <<Tipo de implementación>> = "Java". Dicha definición debe llevar el nombre de la clase, que incluirá el paquete completo al que pertenezca (miApp.trewa.condiciones.ClaseCondiciones), y el nombre de la función a ejecutar. Dicha función debe aparecer definida como *public* en la clase de forma que pueda ser accesible por trew@.

2.- Los métodos asociados a condiciones o acciones deben devolver un valor numérico (objeto java.lang.Integer,...) ,**0** si no se cumple la condición y **1** si se cumple la condición. Los parámetros en este caso deben llevar el siguiente orden y deben ser del tipo que corresponda según la siguiente tabla:

<u>Parámetro en Trew@</u>	<u>Parámetro en método</u>
Id Expediente	java.math.BigDecimal
Id Transición	java.math.BigDecimal
Id Documento permitido	java.math.BigDecimal
Id Expediente en fase	java.math.BigDecimal
Id del procedimiento	java.math.BigDecimal
Fecha	java.sql.Timestamp
Usuario	java.lang.String
Id Fase	java.math.BigDecimal
Id del tipo de documento	java.math.BigDecimal

3.- Los métodos asociados a variables pueden devolver un **java.lang.String** o un **java.util.Map** con los valores de varias variables para mejorar la eficiencia. En el caso de que devuelve un java.util.Map debe estar compuesto por una key que sea un String con el nombre de la variable y en el value otro String con el valor de la variable a sustituir. Los parámetros en este caso deben ser equivalentes a los definidos en los parámetros asociados a la variable, siendo la correspondencia la indicada en la siguiente tabla:

<u>Tipo de parámetro en Trew@</u>	<u>Tipo de parámetro en método</u>
NUMÉRICO	java.lang.Integer
CADENA	java.lang.String
FECHA	java.sql.Timestamp

4.- Los métodos asociados a selectores de plantillas deben devolver un **java.lang.String** con el valor del código de plantilla office asociado al tipo de documento que se desee generar. Los parámetros en este caso deben llevar el siguiente orden y deben ser del tipo que corresponda según la siguiente tabla:

<u>Parámetro en Trew@</u>	<u>Parámetro en método</u>
Id Expediente	java.math.BigDecimal
Id Documento permitido	java.math.BigDecimal
Id Expediente en fase	java.math.BigDecimal
Id del procedimiento	java.math.BigDecimal
Fecha	java.sql.Timestamp
Usuario	java.lang.String
Id del tipo de documento	java.math.BigDecimal

5.- La clase debe aparecer accesible a las clases de Trew@. Esto se puede conseguir de alguna de las formas siguientes (se muestra ejemplo para Tomcat):

Si trewa.jar está a nivel de aplicación

(%TOMCAT%\webapps\AplicacionJSP\WEB-INF\lib\trewa.jar)

En este caso las clases de condiciones y/o variables podrán ubicarse en:

- %TOMCAT%\webapps\AplicacionJSP\WEB-INF\lib (empaquetadas en un jar)
- %TOMCAT%\webapps\AplicacionJSP\WEB-INF\classes (los .class)
- %TOMCAT%\common\lib (empaquetadas en un jar)
- %TOMCAT%\common\classes (los .class)

Si trewa.jar está a nivel de servidor

(%TOMCAT%\common\lib\trewa.jar)

En este caso las clases de condiciones y/o variables podrán ubicarse en:

- %TOMCAT%\common\lib (empaquetadas en un jar)
- %TOMCAT%\common\classes (los .class)

5.- Si se desea que la función a ejecutar en la condición o variable reutilice la interfaz TrAPIUI, la clase a la que pertenezca se tendrá que derivar de la clase

trewa.ext.TrAccesoUI

Heredando de la clase TrAccesoUI se tendrá acceso al método **getApiUI()** a través del cual se podrá obtener la interfaz TrAPIUI y reutilizar la conexión actual.

Además al extender de la clase TrAccesoUI tendremos acceso al método **getNombre()** con el que podemos obtener el nombre de la variable, condición u acción que se esté ejecutando.

A partir de Trew@ v.2 se permite obtener un texto libre indicado en la condición / acción o en la asociación de éstas con la transición, documento permitido o tarea permitida, mediante el método **getTexto()**.

```
import trewa.ext.TrAccesoUI;

public class ClaseVariable extends TrAccesoUI
{
    // Método de variable sin parámetros
    public String metodoVar( java.lang.String val ) throws TrException
    {
        String valRet = "";

        ....
        return str;
    }
}

import trewa.ext.TrAccesoUI;

public class ClaseCondicion extends TrAccesoUI
{
    // Método de condición
    public Integer existenDatos( BigDecimal idExpediente ) throws TrException
    {
        Integer intRet = new Integer(0);

        apiUI = getApiUI();
        if ( apiUI == null ) return intRet;

        ....
        return str;
    }
}
```

En el caso de condiciones complejas, se permite que la condición/es que componen dicha condición compleja devuelvan un String. Con esto podremos comparar el valor de una condición con un valor numérico o una cadena para decidir si la condición se cumple o no. Para ello debemos usar el prefijo **\$\$VAL\$\$** si queremos que se compare por el valor de la condición en lugar de su resultado lógico.

Por ejemplo, suponemos que tenemos definidas dos condiciones, **COND_COMPLEJA** y **COND_SIMPLE**. La **COND_SIMPLE** estará definida como una condición de implementación java que devolverá un String con el valor "2000". La **COND_COMPLEJA** estará definida como una condición compleja de implementación java cuya expresión podría ser "**\$\$VAL\$\$COND_SIMPLE\$\$ > 1000**", esto quiere decir que si el valor de **COND_SIMPLE** es mayor que 1000 se cumplirá la condición.

	<p>Junta de Andalucía Agencia Digital de Andalucía</p>	<p>Trew@ Guía de referencia de JTrApi</p>
---	---	---

3 BIBLIOGRAFÍA Y REFERENCIAS

Referencia	Título	Código
Ref.1	Norma Técnica de Interoperabilidad de Expediente Electrónico	https://www.boe.es/boe/dias/2011/07/30/pdfs/BOE-A-2011-13170.pdf
Ref.2	Trew@	Motor de tramitación de expedientes electrónicos del proyecto w@ndA