

# Trew@ v1.0.0



JUNTA DE ANDALUCÍA  
CONSEJERÍA DE JUSTICIA Y ADMINISTRACIÓN PÚBLICA

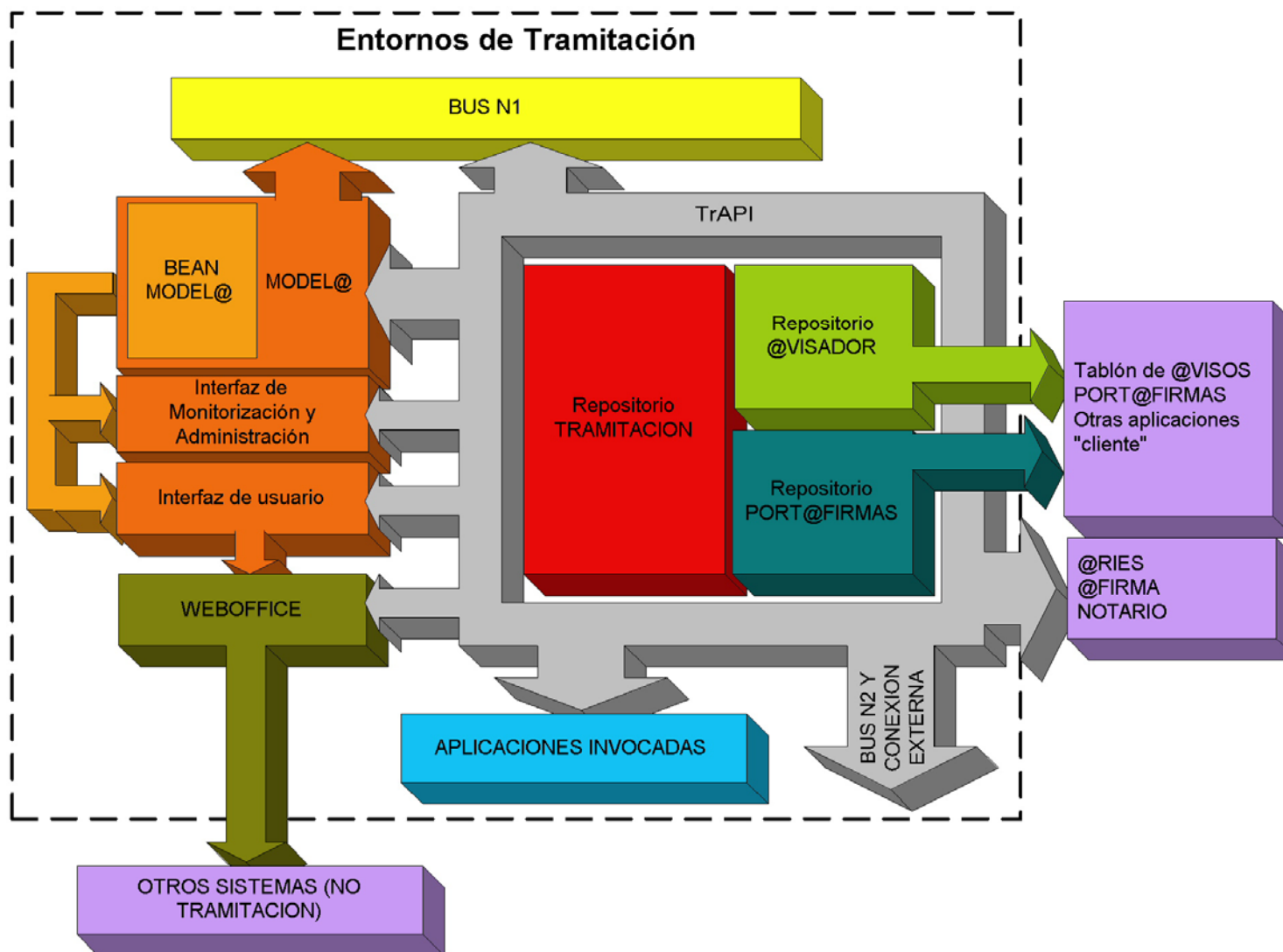
## ¿Por qué?

- Gestionar procedimientos administrativos
- Herramientas workflow
- Necesidad de coordinar actividades y usuarios participantes en el proceso

## Ventajas

- Automatiza tareas y pasos a dar en los procedimientos
- Ayuda a tener procedimientos homogéneos
- Facilita el trabajo ofreciendo a los usuarios participantes qué hacer y en qué momento
- Mantenimiento sin modificar código

# Esquema general



## Requisitos de Implantación

- **Base de datos ORACLE 9i**
- **Servidor de aplicaciones TOMCAT 5.0.18 o 5.0.28 (jdk 1.4.2\_07), TOMCAT 5.5 (jdk 1.5), JBOSS 4.0.0 o 4.0.1 y OC4J 9.0.4.0.0 , etc. para usuarios finales en entornos web (J2EE)**
- **Forms 6i parche 9 (mínimo) para pantallas de administración (preferible en Forms Server, ya que en cliente/servidor están limitadas ciertas funcionalidades)**
- **Servidor Reports 6i parche 9 (mínimo) para generación de escritos basada en reports. Además APACHE de Oracle + módulo PL/SQL o bien IAS para DAD**
- **Cliente Open Office para generación de escritos mediante Web Office**

## Datos Trew@ / Datos “Cliente”

- No gestiona modelo de datos cliente, solo tramitación:
  - “Expediente”: datos principales como nº, título, organismos, interesados, etc.
  - Fases y transiciones y sus fechas
  - Tareas y sus fechas
  - Usuarios asignados al expediente
  - Documentos almacenados en base de datos
- En el modelo de datos de nuestra aplicación cliente debemos evitar los datos que ya existen en Trew@, pero sí podemos completarlo.
- Evitar usar como claves primarias en nuestro modelo de datos las PK que nos facilita Trew@ (impide migraciones)

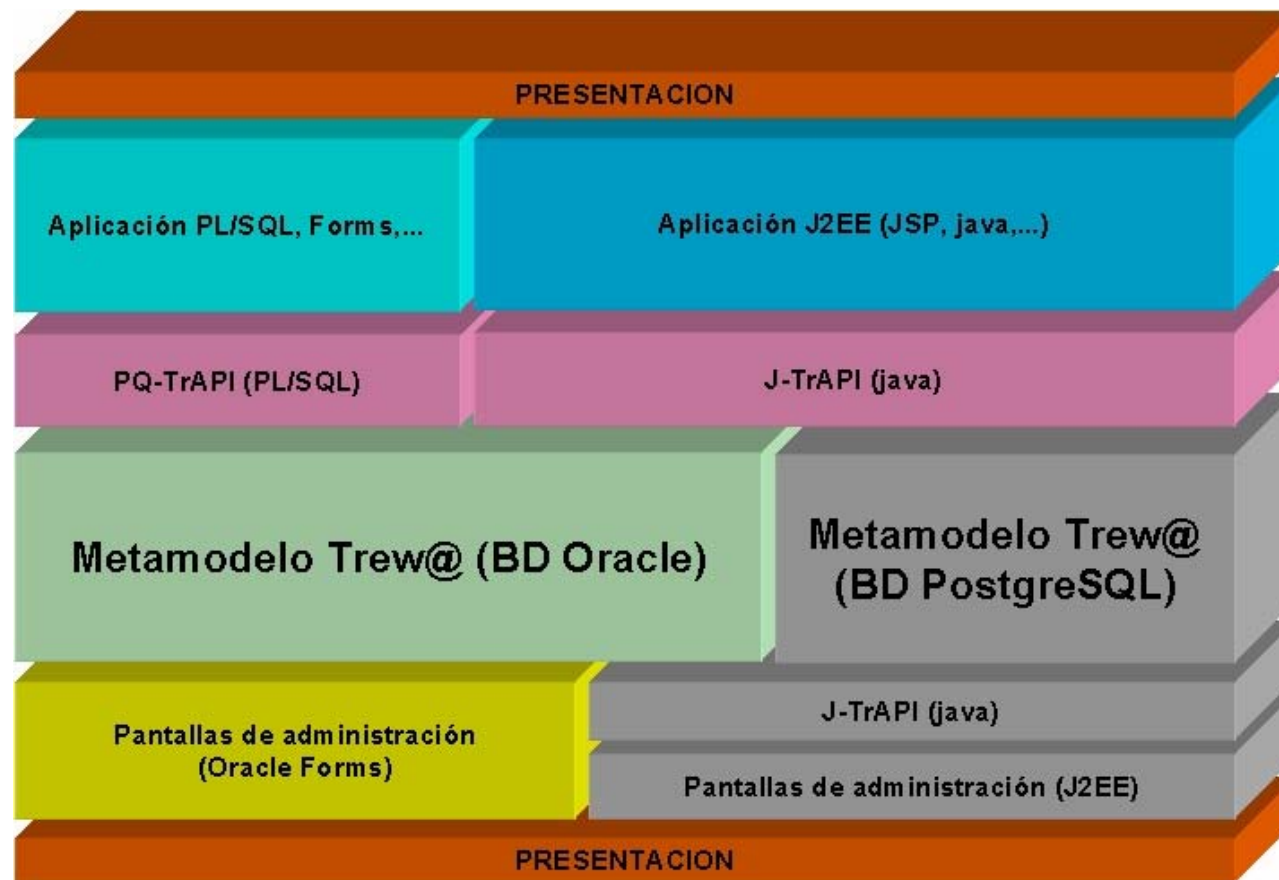
## Pantallas de administración de procedimientos

- En línea
- Implementa funciones de definición (administración y monitor de expedientes más adelante)
- Con “simulador” de tramitación de procedimientos
- Importa / Exporta XML de procedimientos formato Model@
- Exporta XML de expedientes
- Exporta XML de entidades de un sistema para cargar con Model@
- Actualmente en FORMS
- En desarrollo en JSP

## Apis Trew@ v1.0.0

- Única forma de acceder al modelo de datos (otro acceso no garantiza la compatibilidad ni la actualización a posteriores versiones)

## APIs Trew@ Estructura





## APIs Trew@ (características)

- PQ-TrAPI
  - **100% PL/SQL**
  - **Orientada a entornos Oracle**
  - **Organizadas en varios paquetes PL según funcionalidad**
- J-TrAPI
  - **100% Java**
  - **Para aplicaciones J2EE**
  - **Distribuidas en varias interfaces según funcionalidad**
- **Para más detalle consultar la “Guía de referencia”**

## La agenda y el escritorio

- Son pantallas JSP listas para usarse como plantillas en nuestra aplicación
- **Agenda:**
  - Organizador de expedientes del Tramitador.
  - Acceso directo al escritorio
- **Escritorio:**
  - Pantalla de tramitación y gestor de tareas de expedientes
  - Permite ver otros datos relacionados con la tramitación del expediente como transiciones posibles desde la fase actual del expediente, tareas definidas en la fase, caducidades del expediente, tareas asociadas al expediente, gestión de interesados, etc.

## Escritos en Trew@ v1.0.0 (reports)

- Basados en párrafos y variables
- Fusionados en “plantillas” basadas en rdf (Reports de Oracle)
- Se entregan varios rdf. Puede ampliarse
- La fusión de variables exige “visibilidad” del modelo de datos de la aplicación “cliente”
- Model@ y Pantallas de administración

## Componentes propios Trew@ v1.0

- Trew@ incorpora una serie de componentes propios para la integración de los mismos en nuestras aplicaciones si por un aplicativo en particular nos conviene sus funcionalidades en nuestros sistemas.
- Estos componentes no son dependientes de Trew@, por lo que pueden ser instalados y usados de forma separada a la misma.
- Estos componentes son:
  - Port@firmas.
  - @visor
  - Notic@dor: Integrador rápido con la plataforma notificación electrónica Notific@.
  - WebOffice: Editor OpenOffice en línea de documentos.

## Port@firmas

### ¿Que es el Port@firmas?

- Un sistema de almacenamiento y gestión de documentos electrónicos para que un usuario firme digitalmente mediante su certificado de usuario y la plataforma de firma electrónica de la Junta de Andalucía, @firma.
- No es sólo un cliente de firma. Pensada como herramienta de gestión interna, tiene una doble funcionalidad:
  - Ofrecer un portal de firma electrónica genérica de documentos con la plataforma @firma.
  - Habilitar a las aplicaciones actuales el envío de documentos para firma de forma electrónica sin que éstas estén ligadas a una arquitectura web.

## **@visador**

### **Origenes del @visador**

Problemas existentes:

- **Falta de comunicación entre distintos órganos en trámites dependientes.**
- **Sistemas de comunicación proactivos por parte de los usuarios, bien por correo o teléfono, que en ausencia, puede quedar luego en el olvido.**
- **Puede que incluso por desconocimiento, no se sepa a quién hay que avisar.**

Solución:

- **Nace @visador como gestor genérico de avisos que tiene la funcionalidad de enviar “mensajes” entre distintos entornos y/o usuarios.**
- **Integración transparente para los usuarios. Los avisos son generados por las aplicaciones ante determinados eventos y enviados al grupo de usuarios destinatarios definidos en la aplicación.**

## **@visor**

### **Funcionalidades del @visor**

- Envío de correos simples (un correo distinto a cada destinatario) y multiples (el mismo correo a mas de un destinatario).
- Periódicos y configurables en el tiempo con un máximo número de iteraciones.
- Los receptores pueden ser:
  - Usuarios\* de aplicaciones.
  - Grupos\* de usuarios a los que estén suscritos o pertenezcan.
  - Direcciones de correo electrónicas.
- Pantilla HTML de correo personificable por Consejería.
- Recepción de acuses de recibo para seguir la traza de los mismos.
- Visibilidad de estructura de usuarios y grupos de terceras aplicaciones para el avisador a través de vistas de base de datos.
- API's PL/SQL o Java mediante Webservices.
- Tablón de avisos integrado dentro del propio Port@firmas.

## Notific@dor

- El Notific@dor nace como necesidad de integrar con la plataforma de notificación electrónica, Notific@.
- La integración con Notific@ requiere un seguimiento exhaustivo de la notificación y la remesa en la que se remite, y por tanto requiere un modelo de datos paralelo al de Notific@ para poder “preguntar” por el estado de la misma.
- Se debe mantener un control sobre los abonados al servicio de notificación. Requiere procesos periodicos que revisen las entregas de las notificaciones cada hora/dos horas aproximadamente, para hacer un seguimiento correcto de las mismas.
- El Notific@dor presta el servicio de repositorio de notificaciones, reduciendo las informaciones necesarias y omitiendo el concepto de remesa.
- El Notific@dor funciona como Buffer Asíncrono de comunicación con Notific@, permitiendo seguir trabajando normalmente a las aplicaciones ante una caída de red externa.
- El Notific@dor lleva la gestión de abonados suscritos a un servicio así como el seguimiento de las notificaciones realizadas de forma periódica mediante una serie de “Demonios”.



## WebOffice

### ¿Que es WebOffice?

- Esta pensada como una herramienta de ofimática personalizable, que puede ir evolucionando y ampliando con el tiempo en base a las necesidades puntuales.
- En una única herramienta se dispone de una plataforma de edición documental con los siguiente aspectos:
  - Basada en OpenOffice.
  - Lectura de plantillas de documentos originales en base de datos o una ruta web.
  - Sustitución automática y controlada de variables cruzadas con base de datos.
  - Documento editado es guardado nuevamente en la base de datos para poder recuperarlo más adelante o compartirlo con otros usuarios de la aplicación.

**Gracias por su  
atención**



**JUNTA DE ANDALUCÍA**  
CONSEJERÍA DE JUSTICIA Y ADMINISTRACIÓN PÚBLICA