

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)



Powered by
autentia

Hosting Patrocinado por
enREDados.com



[Home](#) | [Quienes Somos](#) | [Empleo](#) | [Tutoriales](#) | [Contacte](#)

Autor: Javier Antonucci
jantoniucci@deloitte.es

Descargar este documento en formato PDF [eclipseJET.pdf](#)

[Firma en nuestro libro de Visitas](#)

Free Eclipse Download

Get Eclipse 3.2, 160 open source plugins, easy to install
www.yoxos.com

Descarga software gratis

Saca el máximo rendimiento de tu PC con Google Pack. Infórmate
pack.google.es

Easy Portlet Building

Intranet Portlets in minutes wizard based. No programming required.
www.kapowtech.com

Altio

Real-time data visualisation, interaction and integration.
www.altio.com

[Anuncios Google](#)

[Anunciarse en este sitio](#)

Manual Básico de Eclipse JET

Introducción

¿Qué es Eclipse JET?

Java Emitter Template (JET) es un subproyecto de Eclipse Modeling Framework (EMF) centrado en simplificar el proceso de generación automática de código (Java, XML, JSP, etc.) a partir de plantillas.

JET funciona a partir de plantillas muy similares a los JSP que, al igual que en esta tecnología, son traducidas a una clase Java para luego ser ejecutadas por medio de una clase generadora creada por el usuario.

¿Para qué sirve?

Sirve para generar automáticamente código fuente (Java, XML, JSP, SQL o cualquier texto) a partir de plantillas.

Es una tecnología que no nos va a dejar sin trabajo a los programadores pero nos va a ayudar mucho a escribir código base sobre el que desarrollar.

Por ejemplo, se podría programar automáticamente los controllers, DAOs y JSP del CRUD (operaciones de crear, leer, actualizar y eliminar) de una entidad a partir de su JavaBean.

¿Cuándo utilizar Eclipse JET?

JET se ajusta mejor a proyectos grandes y estructurados, donde se ha definido un framework (entendido como arquitectura reutilizable) que requiere la programación de extensiones básicamente similares. JET entonces nos puede ayudar a crear el código base sobre el que continuar programando ahorrándonos la escritura de código esencialmente repetitivo. JET puede ayudarnos también a crear plantillas customizadas de JSPs o HTMLs.

¿Cómo utilizar Eclipse JET?

Bueno, eso depende de nuestros recursos e inversión. Lo más bonito es meterlo dentro de un plug-in con interfaz de usuario tipo asistente, pero dependiendo de nuestros conocimientos puede llevarnos las mismas o más horas que el ahorro de utilizar plantillas. Otra forma elegante y más económica es encapsularlo dentro de una tarea Ant. Y finalmente siempre queda el recurso de crear una clase con un main donde metamos

los argumentos en el código del generador.

¿Cómo funciona esto internamente?

Básicamente, se programan las plantillas y el generador. Éste se puede hacer reutilizable para todas las plantillas o especializarse en una o más, a gusto del programador.

Un ejemplo de JET:

1. Convertir el proyecto a un JET Project

File / New / Others / Java Emitter Templates / Convert Projects to JET Projects, seleccionar los proyectos y Finish. El asistente creará una carpeta `templates`.

2. Cambiar los JET Settings

En Project / Project Properties, seleccionar JET Settings e indicar en Source Container el path donde alojamos el código fuente.

3. Crear la plantilla

En la carpeta `templates` crear un fichero (File / New / File) con una extensión terminada en `jet` como por ejemplo `DaoTemplate.javajet`. Atención: generará un error la primera vez porque el fichero está vacío, pero podemos ignorarla. En el mismo incluimos la plantilla, por ejemplo:

```
<%@jet package="com.jas.jet.templates" class="DaoGenerator"
imports="java.util.*" skeleton="generator.skeleton" %>

<%

HashMap argumentMap = (HashMap) argument;

String className = (String) argumentMap.get("className");

String classNameLower = className.toLowerCase();

String searchProperty = (String) argumentMap.get("searchProperty");

String searchPropertyLower = searchProperty==null?
null:searchProperty.toLowerCase();

%>

package com.jas.pruebaSpring.model.dao;

import java.util.List;

import com.jas.pruebaSpring.model.<%= className %>;

public interface <%= className %>Dao {

    <%= className %> find<%= className %>ById(int id);

    <% if (searchProperty!=null) {%>

        List find<%= className %>By<%= searchProperty %>(String <%=
searchPropertyLower %>);

    <% } %>

    void insert<%= className %>(<%= className %> <%= classNameLower %
>);

    void update<%= className %>(<%= className %> <%= classNameLower %
>);

    List getAll<%= className %>s();

    void remove<%= className %>(<%= className %> <%= classNameLower %
```

```
>);  
}
```

Creo que si se conoce la tecnología JSP esta plantilla se entiende bastante bien por sí sola, pero en http://www.eclipse.org/articles/Article-JET/jet_tutorial1.html al final del artículo hay una pequeña guía de referencia de los elementos y directivas de una plantilla JET.

En la carpeta `com.jas.jet.templates` se generará automáticamente una clase `DaoGenerator` que generará la plantilla. Obviamente, este paquete y nombre de clase podemos renombrarlo a nuestro gusto.

En este caso he obligado al generador a implementar utilizar un esqueleto como el que sigue:

```
public class CLASS implements com.jas.jet.GeneratorInterface {  
    public String generate(Object argument) {  
        return "";  
    }  
}
```

Esta interfaz consiste sólo en:

```
public interface GeneratorInterface {  
    String generate(Object argument);  
}
```

De esta forma podré manipular genéricamente a los generadores.

4. Crear el generador

Consiste en una clase que invoque a la plantilla de la siguiente forma:

```
HashMap<String, Object> argumentsMap = new HashMap<String, Object>();  
argumentsMap.put("className", className);  
argumentsMap.put("searchProperty", searchProperty);  
  
GeneratorInterface generator = new DaoGenerator();  
String result = generator.generate(argumentsMap);  
System.out.println(result);
```

Y ya podemos ingeniarnos para hacerle miles de mejoras como parsear la clase generada y saber entonces la carpeta y nombre del fichero donde guardarlo automáticamente.

¿Licencia?

Eclipse JET es parte del proyecto Eclipse y utiliza su licencia, por lo que es Open Source disponible, modificable y comercializable libremente. (todo asesoramiento en esta licencia será bienvenido)

¿Y para qué este manual básico?

Para simplificar el "first touch" y para explicar Eclipse JET a alto nivel.

No pretende ser un tutorial, una guía del usuario ni un manual de referencia. Ellos ya se han escrito y están disponibles en las direcciones que se mencionan en la sección "¿Dónde encuentro más info?"

¿Dónde encuentro más info?

En la Página Oficial del Proyecto EMF es <http://www.eclipse.org/emf> donde encontrarán muy poca información sobre JET. En la sección Documentación se reseña un sólo tutorial (www.eclipse.org/articles/Article-JET/jet_tutorial1.html) de donde se pueden sacar algunas buenas conclusiones. También está disponible el JavaDoc de la API.

Cabe aclarar que a pesar de la falta de documentación es un proyecto totalmente activo aunque muy "eclipsado" por el gran EMF.

A pesar de la falta de documentación, es un framework esencialmente sencillo de utilizar y muy parecido a la tecnología JSP.

¿Cómo pongo en funcionamiento Eclipse JET?

Básicamente en Help / Software updates / Find and install... hay que instalar el "EMF plug-in version 2.2.1" o superior.

Con esto y siguiendo los pasos de la sección "¿Cómo funciona esto internamente?" es suficiente.



[Puedes opinar sobre este tutorial aquí](#)

Recuerda

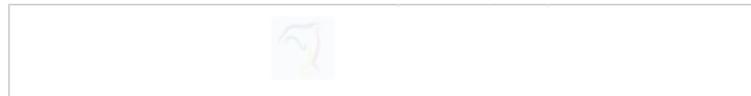
que el personal de [Autentia](#) te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#))

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?

¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

info@autentia.com

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos
Autentia = Soporte a Desarrollo & Formación



[Autentia S.L.](#) Somos expertos en:
J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ..
 y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
<i>e-mail</i>	
	<input type="button" value="Enviar"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto	Descripción
Añadir Scripting a programas Java	Os mostramos como añadir a vuestras aplicaciones soporte de Scripting, gracias a BeansShell. Esto nos permite ejecutar Java generado dinámicamente.
Optimizando código Java con Eclipse Test Performance Tools Platform	En este tutorial vamos a aprender como usar Eclipse Test Performance Tools Platform (TPTP), que nos permite analizar nuestro código
Optimización Java con Eclipse Profiler Plugin	Alejandro Pérez nos enseña como analizar el rendimiento de nuestras aplicaciones con Eclipse Profiler Plugin.
Soporte XML en Eclipse con X-MEN	Alejandro Perez nos enseña como potenciar el entorno eclipse para facilitarnos el trabajo con ficheros xml, gracias al plugin X-MEN
Procesamiento XML en Java con JAXB y WSDP 1.6	Os mostramos como instalar la versión 1.6 de WSDP y como procesar los ficheros XML con uno de sus componentes, JAXB
Transformación de XML y XSL en JSPs	Os mostramos como poder utilizar XML y XSL en JSPs, combinado con el Patrón MVC
Aplicación de Patrones de Diseño en Java	En este tutorial os mostramos como las técnicas avanzadas de diseño (como patrones de diseño) contribuyen a la contrucción de aplicaciones profesionales en Java.
Manejo de Repositorios CVS desde Eclipse	En este tutorial os enseñamos a manejar el repositorio CVS desde la plataforma Eclipse
PMD, Eclipse y NetBeans	Tutorial que describe la instalación y uso de PMD en los entornos de desarrollo Eclipse y NetBeans
Técnicas básicas y poco comentadas en Java	Os mostramos como realizar algunas cosas simples en Java: Formateo de decimales y enteros, gestión de preferencias y comparación entre objetos de nuevas clases

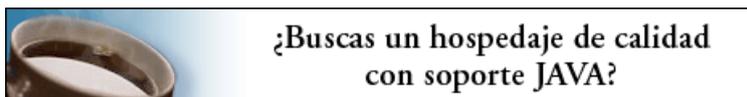
Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)



www.AdictosAlTrabajo.com Optimizado 800X600