

¿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)



Hosting Patrocinado por
enREDados.com


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

	Tutorial desarrollado por: @ Roberto Canales Mora 2003-2005. Si crees que te he ayudado con este tutorial tambien tu me puedes ayudar a mi.. Es muy fácil, mándame un correo con los datos de contacto de tu responsable de Formación y/o Recursos Humanos y es posible que podamos ofrecer un curso interesante a tu organización (y por tanto a tí). Mas detalles... Contacta: rcanales@autentia.com.	
---	---	---

Descargar este documento en formato PDF [wsdp16.pdf](#)

Free JSP Examples

JSP Made Easy With XMLSpy.
Syntax/Editing Help, Free
Download.

Ofertas de trabajo

Oferta y demanda de empleo Abre
las puertas de tu futuro

Trabajos en Canarias

Bolsa de empleo en empresas
líderes Crea tu CV. Te enviamos
las ofertas

¿Harto de tu empleo,

de jefes, de horarios, de atascos,
no llegas a fin de mes? Afiliado

Anuncios Google

Introducción a JAXB con NetBeans

Hay muchas técnicas de manipular ficheros XML en el mundo Java, [como tratamos en tutoriales anteriores](#), y vamos a ver una muy buena opción con JAXB (Java Architecture for XML Binding)

Antes de hacer un ejemplo con esta tecnología, vamos a montar el Kit de desarrollo de Web Services, donde viene incluido (utilizaremos este tutorial como base para otros relacionados con Web Services)

No olvidéis que la primera fuente a consultar siempre, es la original, el tutorial de Sun.

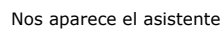
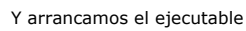
<http://java.sun.com/webservices/docs/1.6/tutorial/doc/index.html>

Y que en www.programacion.com se currar mucho las traducciones (os recomiendo que lo añadáis a vuestras lista de RSS)

<http://www.programacion.com/java/tutorial/jaxb/1>

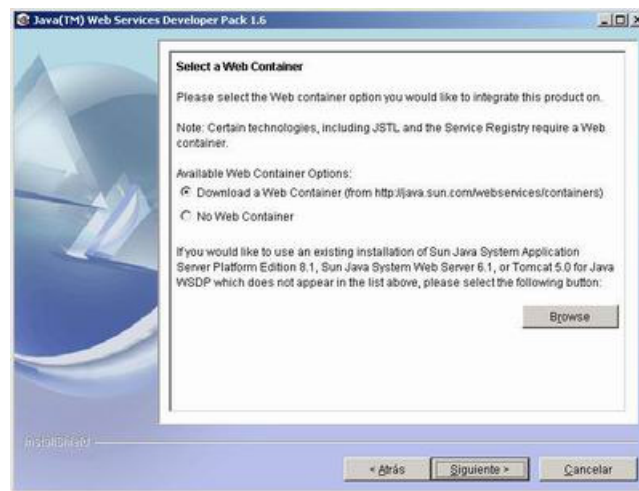
Descarga de WSDP

Lo descargamos del Web de [Sun](#)

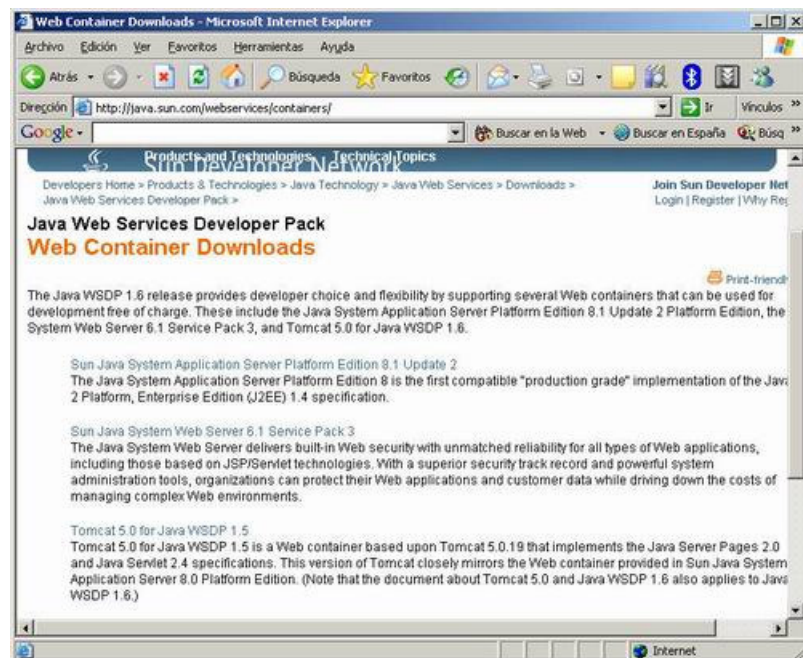




Elegimos la opción de utilizar un Tomcat como contenedor para nuestra aplicaciones y servicios Web



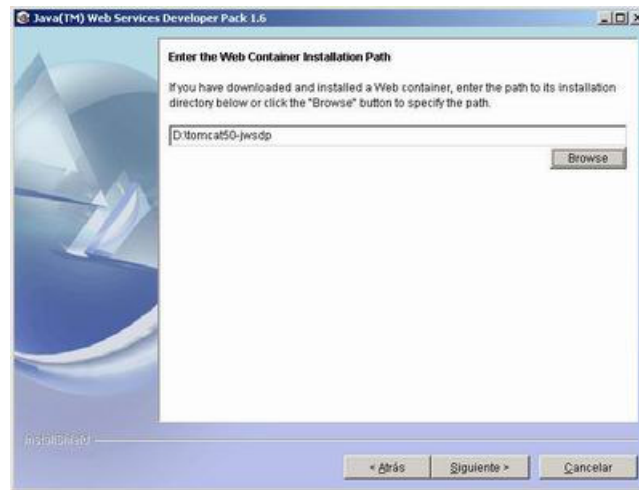
Nos descargamos el Tomcat para WSDP



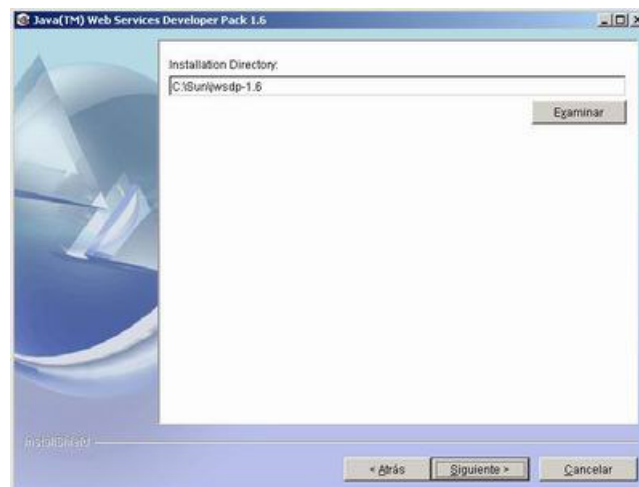
Lo descomprimos un directorio



Ahora seleccionamos el producto recién instalado



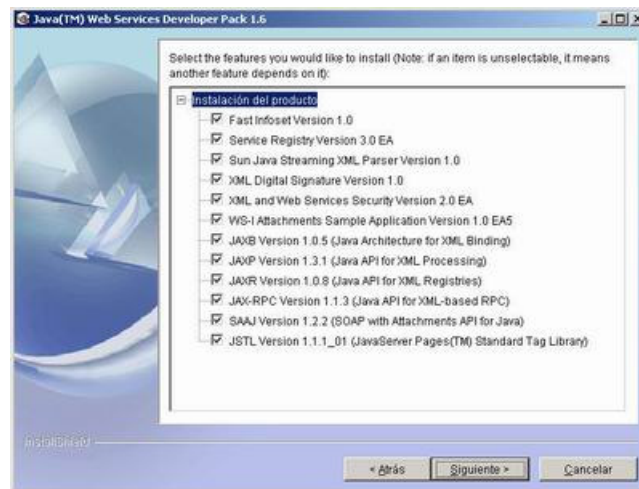
Elegimos el directorio del resto de componentes



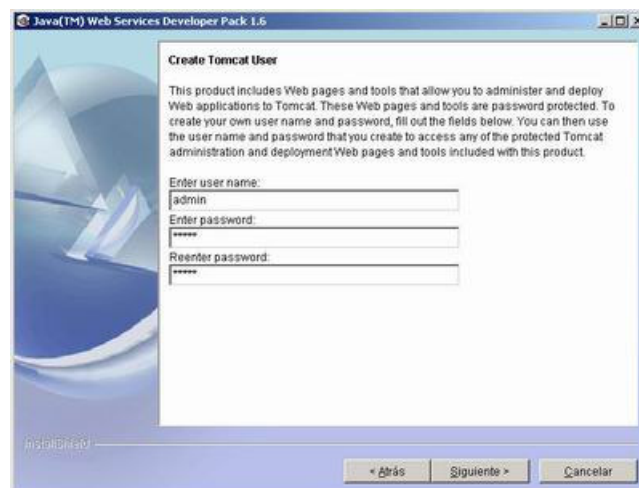
Realizamos la instalación típica



Aunque os muestro las opciones en la personalizada..... que ya tenemos tema para un montón de tutoriales sobre las últimas especificaciones relacionadas con los servicios web (sobre todo debemos estar atentos a la evolución de la seguridad)



Definimos el usuario que usaremos para manejar remotamente el Tomcat



No olvidéis que este usuario hay que darlo de alta en el TOMCAT

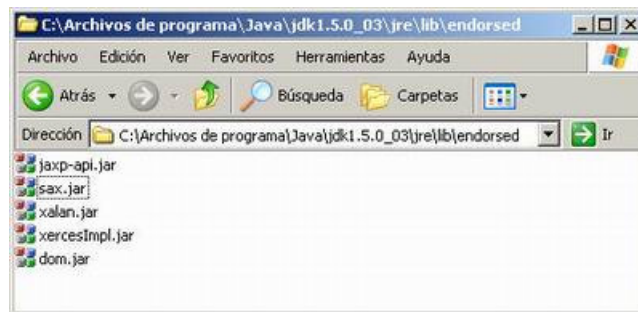
```
<!--
NOTE: By default, no user is included in the
"manager" role required
to operate the "/manager" web application. If you
```

```
wish to use this app,  
you must define such a user - the username and  
password are arbitrary.  
-->  
<tomcat-users>  
<user name="tomcat" password="tomcat"  
roles="tomcat" />  
<user name="role1" password="tomcat"  
roles="role1" />  
<user name="both" password="tomcat"  
roles="tomcat,role1" />  
<user name="admin" password="admin"  
roles="manager,admin" />  
</tomcat-users>
```

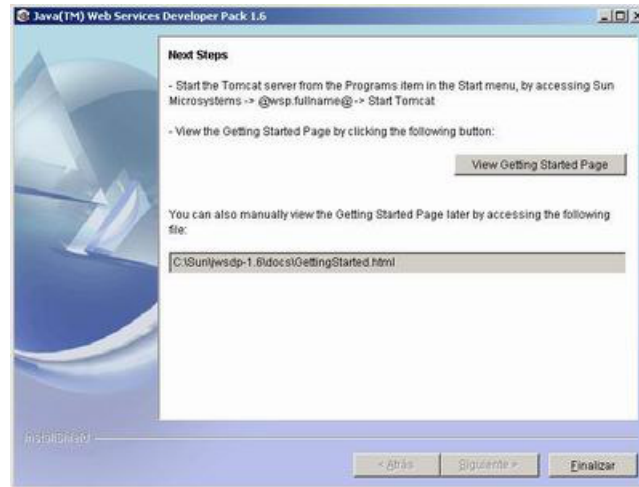
No se os olvide leer las instrucciones finales (lo digo porque al final hacemos ok, ok, ok y no nos funcionan las cosas)



Copiamos los jar al directorio propuesto



Y ahora acabamos

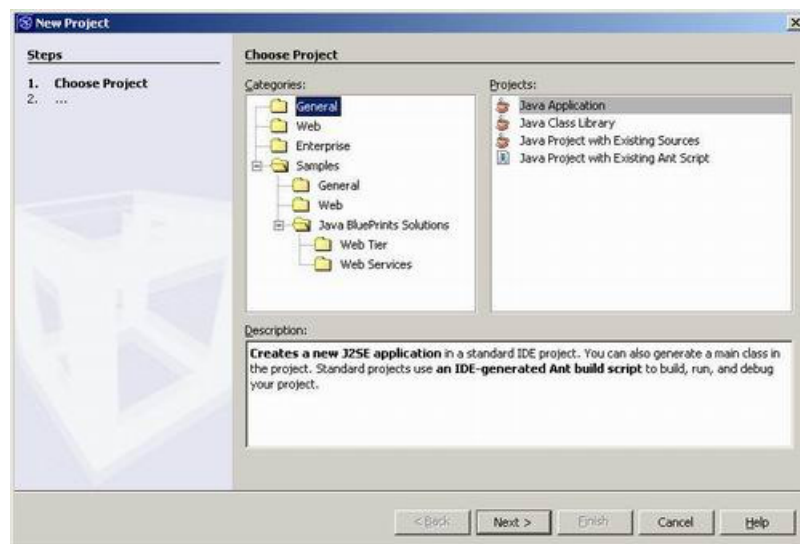


Y al arrancar sobre los iconos creados no podría dar este pequeño error... solo tenemos que ir al directorio de binarios de tomcat y localizar el fichero catalina.bat (actualizando así el acceso directo)

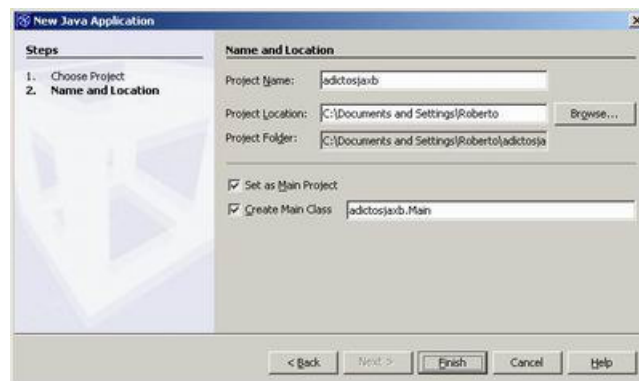


Primer ejemplo de JAXB

Creamos un proyecto dentro de NetBeans 4.1. Será una aplicación Web básica

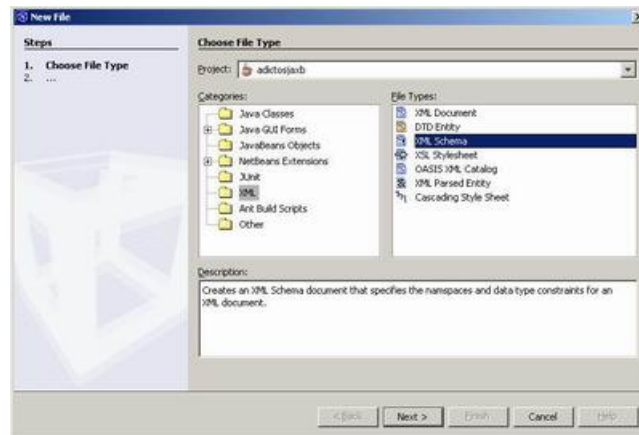


Elegimos el nombre y trayecto

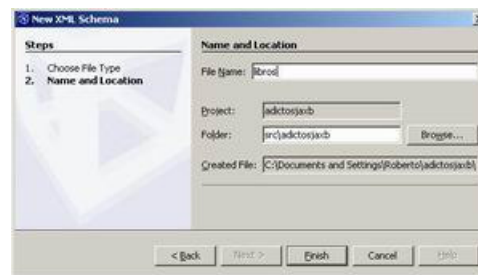


La gracia está en que a partir de un Schema (esquema) XML podemos crear una estructura de clases que nos permitan un acceso cómodo y seguro (verificación de que hacemos lo que debemos en compilación y no en ejecución)

Creamos el esquema



Le damos un nombre (libros.xsd)



```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

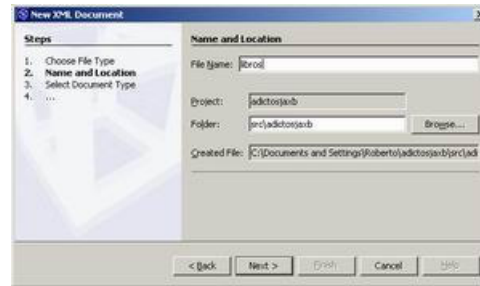
```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.adictosaltrabajo.com/esquemas/libro"
xmlns="http://www.adictosaltrabajo.com/esquemas/libro"
elementFormDefault="qualified">
```

```
<xsd:element name="libro">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="titulo" type="xsd:token"/>
<xsd:element name="autor" type="xsd:token"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
```

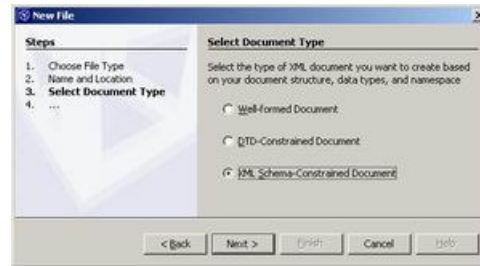
```
<xsd:element name="libros" >
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="libro" maxOccurs="unbounded" />
</xsd:sequence>
</xsd:complexType>
</xsd:element>
```

```
</xsd:schema>
```

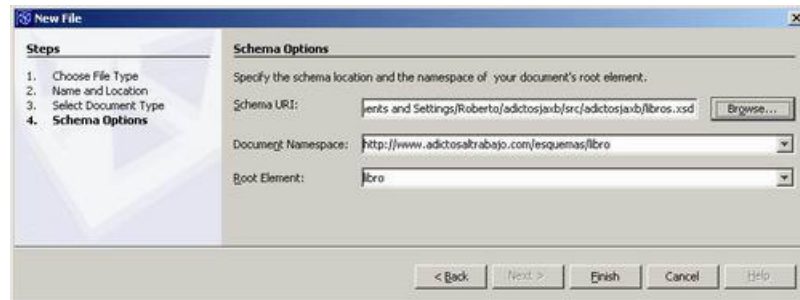
Creamos un fichero XML que use ese esquema



Decimos que se valide contra un esquema



Y le damos los detalles (el fichero lo llamamos libros.xml)



```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<libros xmlns='http://www.adictosaltrabajo.com/esquemas/libro'
xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:schemaLocation='http://www.adictosaltrabajo.com/esquemas/libro
file:/C:/xml/libros.xsd'>
<libro>
<titulo>El fin de una odisea</titulo>
<autor>Desconocido</autor>
</libro>
<libro>
<titulo>Un lugar en la mente de Kerz</titulo>
<autor>Desconocido</autor>
</libro>
</libros>
```

Bueno, hasta ahora no hemos realizado nada raro vamos con la parte específica

Invocamos fichero **xjc.bat** pasando como parámetro el trayecto del fichero xml. También podemos especificar el directorio donde se generarán las clases

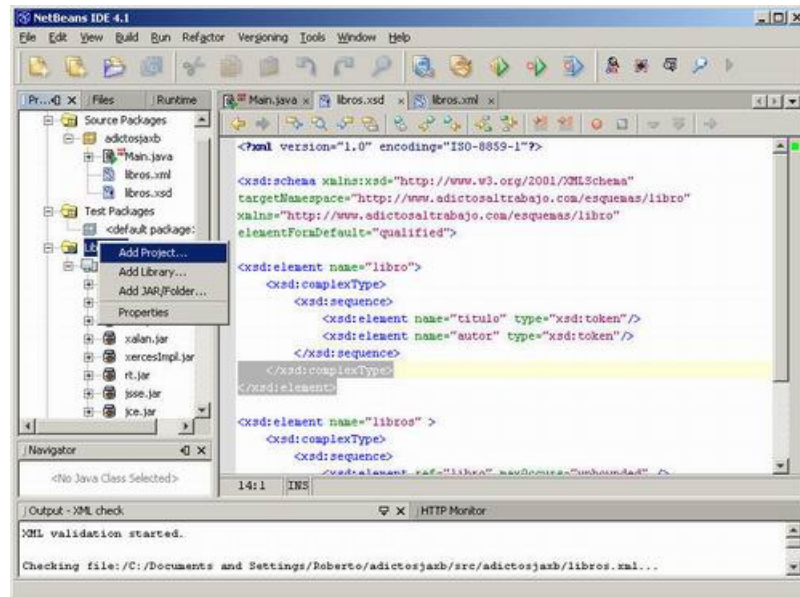
xjc.bat -d directoriofinal ficheros.xsd

```

C:\WINDOWS\system32\cmd.exe
D:\tomcat50-j2ee\j2ee\bin>xcj c:\xml\*.xsd
parsing a schema...
compiling a schema...
com.adictosaltrabajo.esquemas\libro\Libro.java
com.adictosaltrabajo.esquemas\libro\LibroType.java
com.adictosaltrabajo.esquemas\libro\Libros.java
com.adictosaltrabajo.esquemas\libro\LibrosType.java
com.adictosaltrabajo.esquemas\libro\ObjectFactory.java
com.adictosaltrabajo.esquemas\libro\XmlSer
com.adictosaltrabajo.esquemas\libro\XmlSerializer.java
com.adictosaltrabajo.esquemas\libro\impl\run\IdentityHashSet.java
com.adictosaltrabajo.esquemas\libro\impl\run\MarshallerImpl.java
com.adictosaltrabajo.esquemas\libro\impl\run\UnmarshallerImpl.java
com.adictosaltrabajo.esquemas\libro\impl\run\MarshallingEventHandlerImpl.java
com.adictosaltrabajo.esquemas\libro\impl\run\ValidatableObject.java
com.adictosaltrabajo.esquemas\libro\impl\run\ContentHandlerAdapter.java
com.adictosaltrabajo.esquemas\libro\impl\run\ValidatingUnmarshaller.java
com.adictosaltrabajo.esquemas\libro\impl\run\XmlSchemaInfo.java
com.adictosaltrabajo.esquemas\libro\impl\run\UnmarshallerObject.java
com.adictosaltrabajo.esquemas\libro\impl\run\Unmarshaller.java
com.adictosaltrabajo.esquemas\libro\impl\run\XMLSchemaHandler.java
com.adictosaltrabajo.esquemas\libro\impl\run\XMLSchemaHandlerHandler.java
com.adictosaltrabajo.esquemas\libro\impl\run\XMLSerializer.java
com.adictosaltrabajo.esquemas\libro\impl\run\XmlSchemaInfo.java
com.adictosaltrabajo.esquemas\libro\impl\run\UnmarshallerImpl.java
com.adictosaltrabajo.esquemas\libro\impl\run\XMLSchemaHandlerImpl.java
com.adictosaltrabajo.esquemas\libro\impl\run\XMLSchemaHandlerImplImpl.java

```

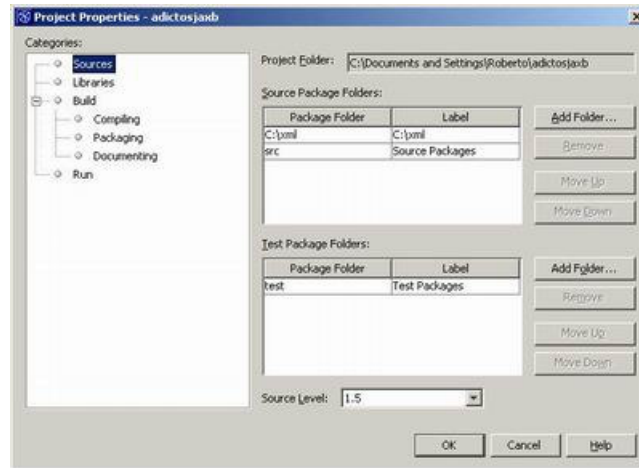
Ahora, en nuestro proyecto le vamos a incorporar los JAR de JAXB



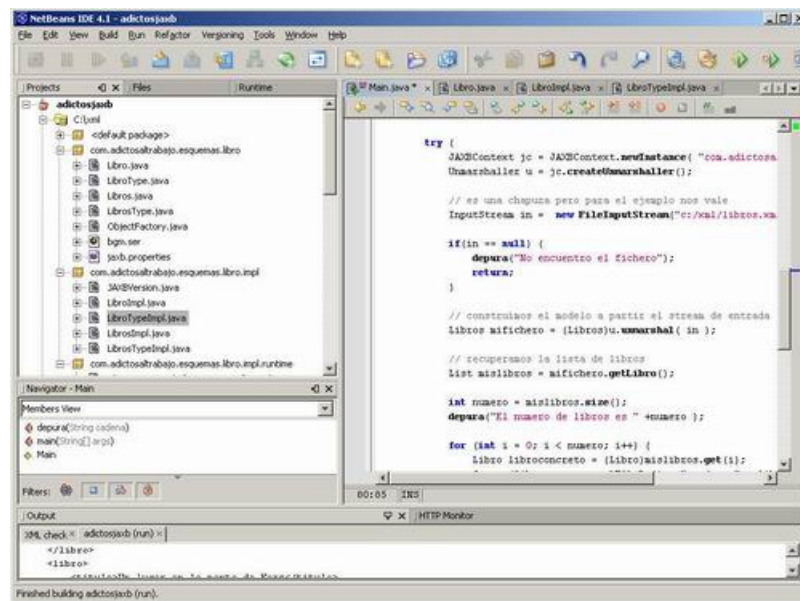
Las elegimos



Y pinchando sobre las propiedades del proyecto, añadimos el trayecto de los ficheros generados automáticamente.



Vemos es aspecto de nuestro proyecto



Y ya solo nos queda escribir el código específico.

```
package adictosjxb;

import com.adictosaltrabajo.esquemas.libro.*;
import com.adictosaltrabajo.esquemas.libro.impl.*;
import javax.xml.bind.*;
import java.io.*;
import java.util.*;

/**
 *
 * @author Roberto Canales
 */
public class Main {

    public static void depura(String cadena) {
        System.out.println("adictosjxb: " + cadena);
    }

    public static void main(String[] args) {

        depura("Empezamos la operación");

        try {
            JAXBContext jc = JAXBContext.newInstance( "com.adictosaltrabajo.esquemas.libro" );
            Unmarshaller u = jc.createUnmarshaller();

            // es una chapuza pero para el ejemplo nos vale

```

```

InputStream in = new FileInputStream("c:/xml/libros.xml");

if(in == null) {
    depura("No encuentro el fichero");
    return;
}

// construimos el modelo a partir el stream de entrada
Libros mifichero = (Libros)u.unmarshal( in );

// recuperamos la lista de libros
List mislibros = mifichero.getLibro();

int numero = mislibros.size();
depura("El numero de libros es " +numero );

for (int i = 0; i < numero; i++) {
    Libro libroconcreto = (Libro)mislibros.get(i);
    depura(libroconcreto.getTitulo() + " - de - " + libroconcreto.getAutor());
}

// ahora añadimos un libro nuevo
LibroTypeImpl nuevoLibro = new LibroTypeImpl();

// establecemos los elementos
nuevoLibro.setTitulo("Domina tu proyecto");
nuevoLibro.setAutor("Anónimo");

// añadimos a la lista
mislibros.add(nuevoLibro);

// vamos a pintar ahora el arbol XML generado para asegurarnos que se ha añadido
Marshaller formateador = jc.createMarshaller();
formateador.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, Boolean.TRUE);
formateador.marshal( mifichero, System.out );

// ahora escribimos a un fichero, en el correcto juego de caracteres
formateador.setProperty(Marshaller.JAXB_ENCODING, "ISO-8859-1");
FileOutputStream salida = new FileOutputStream("c:/xml/salida.xml");
formateador.marshal( mifichero, salida);

} catch(Exception e) // esto es otra chapucilla, no me copieis
{
    depura("el error es " + e.toString());
}
}
}

```

Analizamos el código

Si lo analizamos, veremos que con 4 líneas, hacemos operaciones comunes...

Recuperar el contenido de un fichero:

```

JAXBContext jc = JAXBContext.newInstance( "com.adictosaltrabajo.esquemas.libro" );
Unmarshaller u = jc.createUnmarshaller();

// es una chapuza pero para el ejemplo nos vale
InputStream in = new FileInputStream("c:/xml/libros.xml");

if(in == null) {
    depura("No encuentro el fichero");
    return;
}

// construimos el modelo a partir el stream de entrada
Libros mifichero = (Libros)u.unmarshal( in );

// recuperamos la lista de libros
List mislibros = mifichero.getLibro();

int numero = mislibros.size();
depura("El numero de libros es " +numero );

for (int i = 0; i < numero; i++) {
    Libro libroconcreto = (Libro)mislibros.get(i);
    depura(libroconcreto.getTitulo() + " - de - " + libroconcreto.getAutor());
}

```

Insertar un nuevo nodo

```
// ahora añadimos un libro nuevo
LibroTypeImpl nuevoLibro = new LibroTypeImpl();

// establecemos los elementos
nuevoLibro.setTitulo("Domina tu proyecto");
nuevoLibro.setAutor("Anónimo");

// añadimos a la lista
mislibros.add(nuevoLibro);
```

Volcar el contenido a la pantalla

```
// vamos a pintar ahora el arbol XML generado para asegurarnos que se ha añadido
Marshaller formateador = jc.createMarshaller();
formateador.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, Boolean.TRUE);
formateador.marshal( mifichero, System.out );
```

Y escribir el contenido en un fichero

```
// ahora escribimos a un fichero, en el correcto juego de caracteres
formateador.setProperty(Marshaller.JAXB_ENCODING, "ISO-8859-1");
FileOutputStream salida = new FileOutputStream("c:/xml/salida.xml");
formateador.marshal( mifichero, salida);
```

Conclusiones

La opción de JAXB, junto con JDOM, es una de mis preferidas para manipular de un modo rápido y sencillo ficheros XML.

De todos modos podemos sacar algunas conclusiones importantes:

- Cada vez hay más técnicas para hacer lo mismo.
- Los conocimientos (aparentemente) cada vez tienen un ciclo de vida más corto
- No es tan importante el manipular el fichero XML como definir bien su esquema desde un principio (lo que requiere saber analizar).

Si desea contratar formación, consultoría o desarrollo de piezas a medida puede contactar con

Creatividad Internet

[Autentia S.L.](#) Somos expertos en:
J2EE, C++, OOP, UML, Vignette, Creatividad ..
 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	
e-mail	
	Enviar

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

Nombre Corto

[Introducción a XBRL](#)

[Lectores de RSS](#)

Descripción

Ismael Caballero nos cuenta que es el XBRL, el nuevo estandar de intercambio financiero basado en XML

Os mostramos como instalar y utilizar un lector de RSS (sindicación simple) gratuito, llamado FeedReader, que nos permiten agregar titulares y noticias de nuestros Webs favoritos

[XML básico](#)

Si quieres ver de un modo visual como crear un documento XML, este es tu tutorial. Este es el primero de un conjunto de tutoriales que iremos publicando sobre esta fascinante y amplia tecnología

[Todo está en los libros](#)

Este es un atípico tutorial en nuestro Web donde, a través de la interpretación personal de obras de psicología y estrategia actuales, os invitamos a aprender a apreciar este tipo de libros, lo que seguro contribuirá a vuestra evolución profesional.

[Navegador Mozilla FireFox](#)

En esta ocasión probamos el estado de evolución del navegador gratuito Mozilla FireFox, una verdadera alternativa en el mercado.

[XSLT:FO y FOP](#)

Os mostramos como generar documentos PDF desde XML gracias a FOP y XSLT:FO

[Transformación de XML y XSL en JSPs](#)

Os mostramos como poder utilizar XML y XSL en JSPs, combinado con el Patrón MVC

[Web Services en tu IPAQ](#)

Cesar Crespo nos enseña como programar accesos Web Services desde tu IPAQ en Visual C++ con PocketSOAP, Apache SOAP y Axis

[XML y XSL en Cliente](#)

En este tutorial os enseñamos como formatear documentos XML directamente en vuestro navegador a través de Plantillas XSL. En cursos sucesivos veremos como hacerlo en el servidor, para no crear dependencias con el navegador del cliente.

[SAX, DOM y NetBeans](#)

En este tutorial os vamos a mostrar como manipular documentos XML desde Java y NetBeans

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](#)

