

# Guía de desarrollo de TNTConcept 0.20.X

## Índice de contenido

1. Introducción.....	2
1.1. Referencias.....	2
1.2 Software recomendado.....	2
1.3 Tecnologías empleadas.....	3
2. Eclipse Galileo.....	3
4. MySQL.....	3
5. Apache Tomcat.....	4
5.1 Apache Tomcat y Eclipse IDE .....	5
6. Perfiles Maven para la generación del war.....	6
7. Versionado y cambios sobre la base de datos.....	7

## 1. Introducción

El presente documento tiene como meta ser una guía de referencia que indique qué se debe hacer para poder desarrollar sobre la aplicación de software libre TNTConcept:



Gestión de Eficacia Operativa para PYMES

<http://tntconcept.sourceforge.net/>

### 1.1. Referencias

Para poder hacer modificaciones sobre TNTConcept es **requisito indispensable** que lo tengamos instalado y funcionando en el equipo que vayamos a utilizar para desarrollar. Para ello deberemos utilizar la guía de instalación de TNTConcept, la cual se puede conseguir:

- Si tenemos el código fuente, en el directorio `tntconcept-parent/doc/INSTALL.pdf`
- Si tenemos el instalador, lo descomprimos y lo tenemos en el directorio `doc/INSTALL-{version}.pdf`

No se debe empezar a desarrollar hasta que la aplicación no esté instalada y funcionando.

### 1.2 Software recomendado

Para poder desarrollar bajo TNTConcept se recomienda montar un entorno con el siguiente software:

- JDK 1.6 como versión de la máquina virtual de Java,
- maven 2.1.0 como herramienta de compilación y empaquetado de la aplicación,
- Eclipse como entorno de desarrollo integrado, con el plugin Q4E (ahora llamado IAM),
- Apache Tomcat 6.x como contenedor web donde desplegar la aplicación,
- MySQL 5.1.31 como sistema gestor de bases de datos donde la aplicación almacenará la información.

A lo largo del documento se darán algunas pautas de cómo utilizar este software para poder desarrollar TNTConcept.

## 1.3 Tecnologías empleadas

TNTConcept es una aplicación web que utiliza, entre otras, las siguientes tecnologías:

- Java Server Faces (JSF) con la librería Tomahawk como implementación del patrón MVC.
- Hibernate como ORM y motor de persistencia.
- Jasperreports como motor de generación de informes en PDF y en otros formatos.
- ACEGI como encargado de gestionar la seguridad de la aplicación.
- Spring como elemento de unión entre componentes.
- Javamail para permitir el envío de correos electrónicos.
- Junit como framework con el que poder generar casos de prueba.
- Log4j como librería de generación de trazas.

## 2. Eclipse Galileo

Se recomienda utilizar este entorno de desarrollo (IDE) como herramienta con la cual poder realizar cambios sobre TNTConcept.

Sobre dicho entorno se deberán instalar los plugins de maven y de subversion, los cuales nos permitirán:

1. poder sincronizar con el repositorio de subversion y
2. que el propio IDE genere el classpath en base a las dependencias definidas en el pom.xml de nuestro proyecto.

Hay un tutorial donde se explica cómo instalar Galileo con estos plugins desde cero:

<http://adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=galileo>

## 4. MySQL

Es necesario crear un esquema propio para TNTConcept y además es muy recomendable crear un usuario específico con todos los privilegios sobre ese esquema.

Para ello utilizaremos el MySQL Administrator (<http://dev.mysql.com/downloads/gui-tools/5.0.html>), al que accederemos como root. El esquema lo crearemos en el apartado 'Catalogs' especificando un nombre (por ejemplo, tntconcept y con el collation por defecto a utf8\_spanish\_ci) que permita identificar qué aplicación es la que hace uso de él.

Una vez creado el esquema se recomienda crear un usuario. Para ello entraremos en el apartado 'User administration', crearemos un usuario (por ejemplo, tntconcept) y le asignaremos una contraseña. Para asignarle permisos sobre el esquema creado con anterioridad:

- Pincharemos sobre la pestaña 'Schema privileges'
- Seleccionaremos el esquema creado anteriormente. Veremos que la columna de la izquierda (llamada 'Assigned privileges') aparece vacía mientras que la columna de la derecha (llamada 'Available privileges') aparece llena.
- Seleccionaremos los privilegios de la columna de la derecha y los moveremos a la columna de la izquierda utilizando los botones que aparecen entre medias de ambas columnas.
- Tras ello pulsaremos sobre el botón 'Aplicar'

Una vez realizados estos pasos tendremos tanto el esquema de base de datos como el usuario con el que poder trabajar sobre esa base de datos. Necesitaremos todos estos datos para configurar, en tomcat, la conexión con la base de datos.

## 5. Apache Tomcat

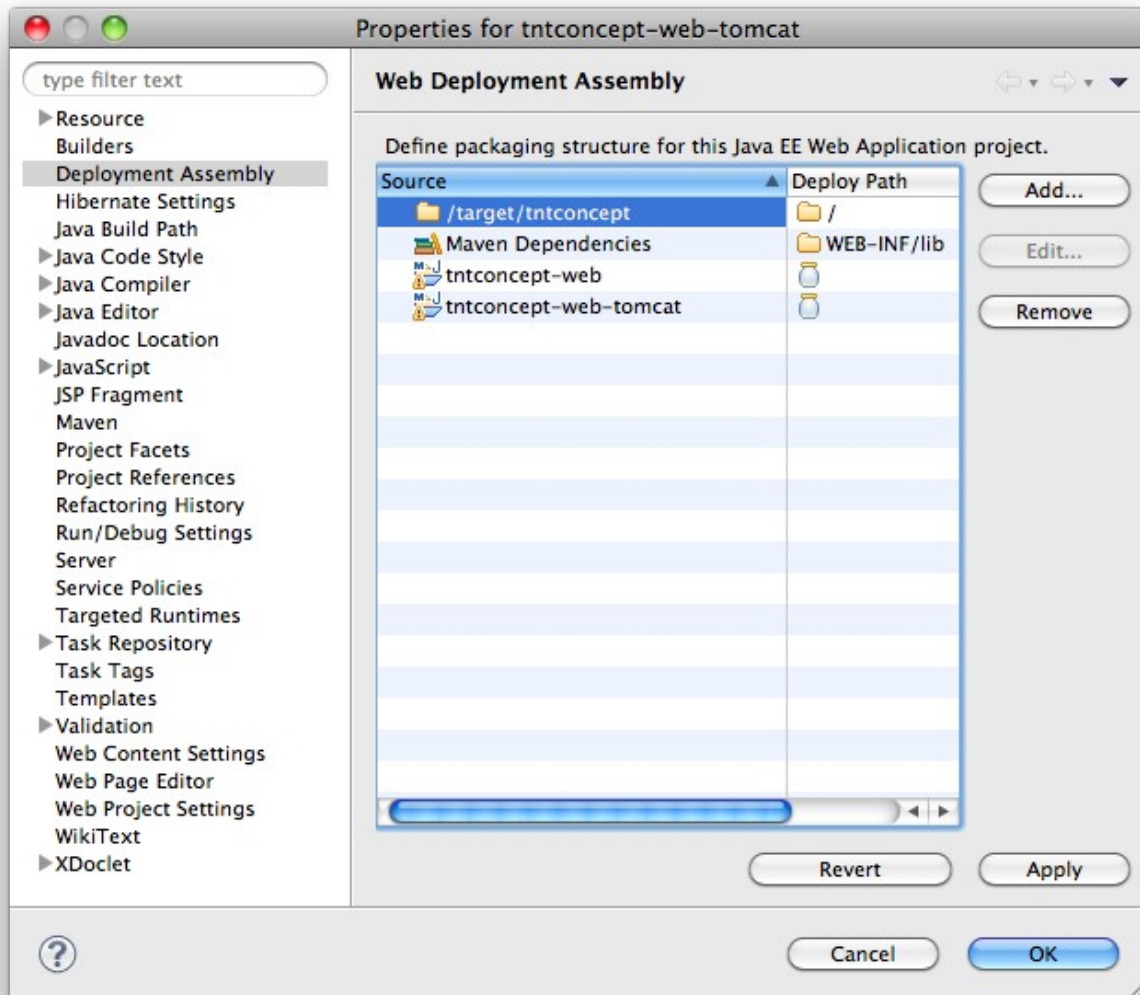
Para configurar tomcat es necesario hacer lo siguiente:

- Configurar la conexión contra la base de datos en el server.xml. Para ello es necesario utilizar el esquema, el usuario y la contraseña especificados en el apartado anterior.
- Indicar cuál va a ser el directorio de configuración de TNTConcept en el fichero server.xml
- Instalar el jar con el conector de la base de datos mysql

El cómo hacer todo esto viene explicado en la guía de instalación de TNTConcept mencionada en el apartado 1.1.

## 5.1 Apache Tomcat y Eclipse IDE

Para poder trabajar y desplegar correctamente el proyecto desde una instancia de tomcat desde el propio eclipse será necesario realizar los siguientes pasos :



1- Modificación del path donde tomcat coje el directorio a desplegar. Para ello pulsamos botón dcho sobre el proyecto tntconcept-web-tomcat → properties

Indicamos con la propiedad AssemblyDevelopment el path correspondiente en este caso tntconcept-web-tomcat/target/tntconcept.

2- Agregamos un nuevo servidor desde la pestaña server y añadimos el modulo tntconcept-web-tomcat.

3- Editar el fichero server.xml para definir el datasource. Como veremos a continuación contiene algunas diferencias con respecto al proceso de instalación, quedando así:

```
<Context docBase="tntconcept-web-tomcat" path="/tntconcept-web-tomcat" reloadable="true"
source="org.eclipse.jst.jee.server:tntconcept-web-tomcat">

  <Manager pathname=""/>

  <Environment name="TNTConceptConfigDir" type="java.lang.String" value="/etc/tntconcept"/>

  <Resource name="jdbc/TNTConcept" auth="Container" type="javax.sql.DataSource" maxActive="100"
    maxIdle="30" maxWait="10000" driverClassName="com.mysql.jdbc.Driver"
    username="tnt" password="tnt" url="jdbc:mysql://localhost:3306/tntconcept?autoReconnect=true"/>

</Context>
```

Básicamente la diferencia esta en la cabecera del tag Context donde le indicamos el docbase y el path adecuados además de la propiedad reloadable y source necesarias para el correcto despliegue de la aplicación desde la instancia de tomcat en Eclipse.

## 6. Perfiles Maven para la generación del war

Tntconcept esta preparado para poder correr tanto en Tomcat como en Jboss.

En la guía de instalación se dan las pautas necesarias para la configuración de cada uno de los servidores pero cuando estamos trabajando en el entorno de desarrollo es importante saber que cada vez que generamos el war con maven por defecto se genera un war preparado para desplegar en Tomcat.

Si quisieramos usar Jboss como contenedor en nuestro desarrollo sería necesario indicarselo a maven ejecutando el comando de maven con el perfil de jboss de la siguiente manera:

```
mvn groupId:artifactId:goal -P jboss-profile
```

Si por el contrario quisiesemos usar como contenedor Tomcat se lo indicamos de la siguiente manera

```
mvn groupId:artifactId:goal -P tomcat-profile
```

## 7. Versionado y cambios sobre la base de datos

El sistema de versionado de TNTConcept se basa en que los cambios que impliquen modificaciones en base de datos suponen un incremento en el número de versión mientras que los cambios que no afecten a la base de datos suponen un incremento de revisión. Pongamos ejemplos:

- En su momento se liberó la versión 0.15.0. Si sobre esa versión se hiciesen cambios que afectasen a la estructura de la base de datos (nuevas tablas, nuevas relaciones, eliminación de las mismas, etc) habría que incrementar el número de versión y generar la versión 0.16.0.
- Si sobre esa versión 0.16.0 se hacen cambios que no afectan a la base de datos habría que incrementar el número de revisión. De este modo se generaría la versión 0.16.1.
- Si sobre esta versión 0.16.1 se volviesen a realizar cambios que no afectan a la estructura de base de datos se generaría una versión 0.16.2.
- Si los cambios a realizar sobre la versión 0.16.2 implican cambios en la base de datos se debe generar una versión 0.17.0.

TNTConcept dispone de un componente encargado de realizar las tareas de migración cuando se ha generado una nueva versión que implica cambios en base de datos. El funcionamiento de dicho componente está explicado en el apartado 5 del documento de instalación de TNTConcept.

Si quisiéramos realizar cambios en TNTConcept que implicasen cambios en base de datos deberíamos hacer lo siguiente:

- Crear un fichero de script llamado 'upgrade.sql'
- Dentro de este script añadir todas las sentencias SQL que se encarguen de modificar la base de datos.
- Al final del fichero incrementar el número de versión de TNTConcept.

Por ejemplo, si nos encontrásemos desarrollando la versión 0.15.2 y quisiéramos hacer cambios que implicasen modificaciones sobre la base de datos (con lo cual la versión que contendría los cambios sería la 0.16.0) deberíamos:

- Crear un fichero llamado 'upgrade.sql' en la siguiente ruta:

```
tntconcept-parent/tntconcept-core/src/main/resources/com/autentia/tnt/sql/mysql/0.15
```

- Añadir en el nuevo fichero todas las sentencias SQL que necesitemos para modificar la base de datos y dejarla en el estado que deseemos.
- Tras añadir todas las consultas deberemos insertar la siguiente sentencia:

```
update Version set version='0.16';
```

De este modo al rearrancar TNTConcept el componente encargado de realizar la migración lanzará los scripts de base de datos y actualizará el número de versión.