

# Installing JIRA on Tomcat 6.0

This section describes how to install JIRA on [Tomcat 6.0](#), a popular open-source server from the Apache Jakarta project.

Tomcat can be downloaded from the [Apache site](#).

## Note:

The JIRA 'Standalone' download is JIRA preconfigured with a copy of Tomcat 5.5.20. If you have JIRA Standalone, you don't need to follow these steps.

## Warning:

All versions of Tomcat [effectively leak memory by caching JSPs](#), which can [result in OutOfMemoryErrors](#) if large pages JIRA (eg. RSS or Excel) are requested. There is a [flag](#) you should set to disable this caching.

## 1. 1. Unpack JIRA

Unzip the JIRA WAR (Webapp ARchive) distribution. A new directory containing JIRA will be created, hereafter referred to as `$JIRA_HOME`

## 2. 2. Configure JIRA

JIRA needs to be told what type of database you'll be using. The database is specified in `$JIRA_HOME/edit-webapp/WEB-INF/classes/entityengine.xml`. Locate the `<datasource>` tag near the bottom, and change the **field-type-name** attribute value:

```
<datasource name="defaultDS"
  field-type-name="hsql"
  schema-name="PUBLIC"
  helper-class="org.ofbiz.core.entity.GenericHelperDAO"
  check-on-start="true"
  use-foreign-keys="false"
  use-foreign-key-indices="false"
  check-fks-on-start="false"
  check-fk-indices-on-start="false"
  add-missing-on-start="true">
  <jndi-jdbc jndi-server-name="default"
    jndi-name="java:comp/env/jdbc/JiraDS" />
</datasource>
```

Possible values include **cloudscape**, **db2**, **firebird**, **hsql**, **mckoidb**, **mysql**, **mssql**, **oracle**, **postgres72**, **sapdb**, and **sybase**

For Postgres 7.3+ and DB2 you also need to set a **schema-name** attribute (see the [PostgreSQL](#) and [DB2](#) pages).

Also in `entityengine.xml`, ensure the `<transaction-factory>...</transaction-factory>` tag contains:

```
<transaction-factory class="org.ofbiz.core.entity.transaction.JNDIFactory">
  <user-transaction-jndi jndi-server-name="default" jndi-name="java:comp/env/User" />
  <transaction-manager-jndi jndi-server-name="default" jndi-name="java:comp/env/User" />
</transaction-factory>
```

More details on JIRA's database access layer are available on the [EntityEngine configuration page](#).

## 3. 3. Build JIRA

Now build JIRA by typing **build** (Windows) or **./build.sh** (Unix) on the command line in the

`$JIRA_HOME` directory. This will produce the deployable WAR file in the `$JIRA_HOME/dist-tomcat/tomcat-6` directory.

#### 4. 4. Update Tomcat Libraries

Tomcat does not come with some libraries required to run JIRA. To fix this, download [jira-jars-tomcat6.zip](#) (1.5Mb), and copy the contained jars to Tomcat's `lib/` directory.

#### 5. 5. Configure Tomcat

A JIRA 'context' now needs to be set up in Tomcat. To do this:

1. Make a `Catalina/localhost/` directory in Tomcat's `conf/` directory.
2. Copy `dist-tomcat/tomcat-6/jira.xml` from the built JIRA distribution to your Tomcat's `conf/Catalina/localhost/` directory.
3. Customise the copied `jira.xml` as follows:

```
<Context path="/jira" docBase="path/to/atlassian-jira-3.12.3.war">
  <Resource name="jdbc/JiraDS" auth="Container" type="javax.sql.DataSource"
    username="sa"
    password=""
    driverClassName="org.hsqldb.jdbcDriver"
    url="jdbc:hsqldb:path/to/database/jiradb/"
    minEvictableIdleTimeMillis="4000"
    timeBetweenEvictionRunsMillis="5000"/>

  <Resource name="UserTransaction" auth="Container" type="javax.transaction.UserTran
    factory="org.objectweb.jotm.UserTransactionFactory" jotm.timeout="60"/>
  <Manager pathname="" />
</Context>
```

The paths (denoted as `path/to/`) will be correct by default, assuming you want to deploy the `.war` from the `dist-tomcat/tomcat-6` directory.

#### Note:

If you are *not* using `hsqldb`, make sure you comment out the `minEvictableIdleTimeMillis` and `timeBetweenEvictionRunsMillis` params, or JIRA will run slower than normal.

If you are installing in Windows, make sure that the paths you specify for the location of the WAR file and database are full paths with drive letters (e.g. `c:\yourdb\tomcatdb`). **N.B.** the last part of the path is the name of the database and is not a directory.

The above example assumes you are using [hsql](#) (an in-memory database — a good choice for a first attempt). Here is an example using MySQL:

```
<Context path="/jira" docBase="path/to/atlassian-jira-3.12.3.war">
  <Resource name="jdbc/JiraDS" auth="Container" type="javax.sql.DataSource"
    username="jirauser"
    password="mypassword"
    driverClassName="com.mysql.jdbc.Driver"
    url="jdbc:mysql://localhost/jiradb?autoReconnect=true&useUnicode=true&chara
  <Resource name="UserTransaction" auth="Container" type="javax.transaction.UserTransac
    factory="org.objectweb.jotm.UserTransactionFactory"
    jotm.timeout="60"/>
  <Manager pathname="" />
</Context>
```

Notice the lack of `minEvictableIdleTimeMillis` and `timeBetweenEvictionRunsMillis` parameters - those should **only** be used with `hsqldb`. If using a different database than `hsqldb`, remember to update the **field-type-name** ([see above](#)) and copy the JDBC driver jar to `lib/` (see the [database configuration guide](#)).

## 6. Modify Tomcat server.xml

In order for JIRA to correctly display internationalised characters in user and group names you need to modify your Tomcat distributions `conf/server.xml` file. You need to set the property `useBodyEncodingForURI="true"` within the connector definition for your http protocol. The connector block should look very much like this:

```
<Connector port="8080" protocol="HTTP/1.1"
  connectionTimeout="20000"
  redirectPort="8443"/>
```

You should modify the block to contain the addition of the `useBodyEncodingForURI` property:

```
<Connector port="8080" protocol="HTTP/1.1"
  connectionTimeout="20000"
  redirectPort="8443" useBodyEncodingForURI="true"/>
```

### Note:

Because you must define this property in at the connector level this setting will effect all web-applications you have deployed under the connector. This should not adversely effect the other web-applications but please be aware of this. JIRA will run fine without this property set but you will run into issues if a user or group is created which contains international characters. It is best to set this property to true.

## 7. Fix Tomcat memory settings

Tomcat has a [memory leak](#) where large JSP page requests can fill up memory. To avoid this, edit Tomcat's `bin/setenv.sh` (create it if it does not exist) and set:

```
export CATALINA_OPTS="$CATALINA_OPTS -Dorg.apache.jasper.runtime.BodyContentImpl.LIMIT_BUFFER_SIZE=2048
```

or when [installed as a Windows service](#), run:

```
tomcat5 //US//JIRA ++JvmOptions="-Dorg.apache.jasper.runtime.BodyContentImpl.LIMIT_BUFFER_SIZE=2048
```

For other environments, and for more info on memory settings, see the [memory settings](#) page.

## 8. Start Tomcat

JIRA should now be ready to run in Tomcat. To start using JIRA, first start (or restart) the Tomcat server with Tomcat's `bin/startup.(sh|bat)` scripts, and point your browser to <http://localhost:8080/jira>

You should now see the [Setup Wizard](#), which will take you through the brief setup procedure.

## 9. Troubleshooting

It is easy to make a mistake in this process, and even more so if you are trying to connect to a database other than hsqldb. First, check that you have followed the process described above:

- If you are using an external database (not hsqldb), have you set the **field-type-name** attribute in `$JIRA_HOME/edit-webapp/WEB-INF/classes/entityengine.xml`? ([step 1](#))
- Have you previously started JIRA with an incorrect **field-type-name** value? If so, the database schema would have been created incorrectly.
- If you have made changes to `$JIRA_HOME/edit-webapp/WEB-INF/classes/entityengine.xml` ([step 2](#)) and re-run the build script ([step 3](#)), but your changes are not being picked up, delete the Tomcat `webapps/jira` directory, then restart JIRA. It would seem that in some circumstances Tomcat does not correctly re-expand the web application.
- Have you copied the extra Tomcat jars ([step 4](#))? Check if you have `common/lib/objectweb-datasource-1.4.3.jar` present.
- If using an external database, did you copy the JDBC driver jar to `common/lib/` ([step 5](#))?
- Is the path to the `.war` file in `conf/Catalina/localhost/jira.xml` correct?

- Have you copied the .war file to Tomcat's **webapps/** directory? This is almost guaranteed to cause problems — please move it elsewhere, and delete any JIRA subdirectories created in **webapps/** from previous Tomcat starts.
- Have you configured JIRA centrally in **conf/server.xml** instead of in **conf/Catalina/localhost/jira.xml**? This is fine, but then be sure you *don't* also have a **conf/Catalina/localhost/jira.xml** present.
- The log files are usually vital to debugging problems. On Windows, these will appear in the console window that loads when running `startup.bat`, or in one of the log files in the `logs/` directory. On Linux/Unix, logs will appear in a log file in `logs/`, usually `logs/*` (not just `logs/catalina.out!`). Check the log file for errors after startup.
- If you experience high memory usage / memory leaks (eg `OutOfMemoryError`), you may wish to set the system property `-Dorg.apache.jasper.runtime.BodyContentImpl.LIMIT_BUFFER=true` in **setenv.sh** / **setenv.bat**. This property is only valid for Tomcat 5.5.15 and later. For more information please see [JIRA-10145](#).
- If you receive exceptions related to logging, please ensure that **commons-logging-1.0.4.jar** and **log4j-1.2.7.jar** are present in Tomcat's `/lib/` directory. Also ensure that these files are not present in Tomcat's `webapps/jira/WEB-INF/lib` directory.

If you're stuck, please raise a [support request](#), and attach your logs, configuration files, plus anything else relevant, and we'll get back to you as soon as possible. If you have a general question, please try the [jira-user mailing list](#) (which Atlassian staff monitor).

## 10. User-contributed notes

Do you have experiences to share with Tomcat 6.0.x and JIRA? We welcome your thoughts. Please see the [user-contributed Tomcat 6.0.x notes](#).