

# Connecting JIRA to Oracle

These instructions will help you connect JIRA to an [Oracle 10g or 11g](#) database. Oracle 9i is no longer a supported database for use with JIRA and the 11.2.x drivers from Oracle do not support 9i.

## 1. Before You Begin

### 1.1 Check the Compatibility of Your Oracle Server with JIRA

Please note that a number of the Oracle server versions cannot be used with JIRA or are inherently unstable. The known issues with Oracle servers are as follows:

- We **recommend** that you **avoid** using version 10.2.0.3 of the Oracle server. Oracle server version 10.2.0.3 has been noted to produce occurrences of error ORA-01461. The [Oracle Support](#) site has further details on this Oracle server issue, although you will need an Oracle support account to access this site. JIRA customers with this problem have reported that upgrading to Oracle server version 10.2.0.4 resolves the issue.

### 1.2 Are You Migrating JIRA to Another Server?

If you are [Migrating JIRA to Another Server](#), create an export of your data as an [XML backup](#). You will then be able to transfer data from your old database to your new database, as described in [Switching databases](#).

### 1.3 Shut Down JIRA

#### On this page:

- 1. Before You Begin
  - 1.1 Check the Compatibility of Your Oracle Server with JIRA
  - 1.2 Are You Migrating JIRA to Another Server?
  - 1.3 Shut Down JIRA
- 2. Configure Oracle
- 3. Copy the Oracle JDBC Driver to Your Application Server (JIRA WAR Only)
- 4. Configure Your JIRA Server to Connect to Your Oracle Database
  - 4.1 Connecting JIRA to Oracle Using the JIRA Configuration Tool
  - 4.2 Connecting JIRA to Oracle Manually
- 5. Next steps
- Installation notes

## 2. Configure Oracle

1. Ensure that you have a database instance available for JIRA (either create a new one or use an existing one).
2. Within that database instance, create a user which JIRA will connect as (e.g. `jiradbuser`).  Remember this database user name, as it will be used to configure JIRA's connection to this database in subsequent steps.

 When you create a user in Oracle, Oracle will create a 'schema' automatically.

 When you create a user, the tablespace for the table objects must be specified.

```
create user <user> identified by <user_pass> default tablespace
<tablespace_name> quota unlimited on <tablespace_name>;
```

3. Ensure that the user has the following permissions:

```
grant connect to <user>;
grant create table to <user>;
```

 If you use GreenHopper 5.8 or later, also ensure that you specify the following permissions too:

```
grant create sequence to <user>;
grant create trigger to <user>;
```

### 3. Copy the Oracle JDBC Driver to Your Application Server (JIRA WAR Only)

 Skip this step if you are using [JIRA Standalone](#) distribution. JIRA Standalone includes the Oracle JDBC driver.

1. Download the Oracle JDBC driver (from [Oracle's site](#)).
2. Add the appropriate Oracle JDBC driver jar (`ojdbc5.jar` for JDK 1.5, `ojdbc6.jar` for JDK 1.6) to the `lib/` directory.

Please note that a number of the Oracle JDBC driver versions cannot be used with JIRA or are inherently unstable. The known issues with Oracle drivers are as follows:

- We **recommend** that you use the [11.2.x version](#) of the driver for all versions of Oracle (it is backwards compatible). Many other versions of the driver have been noted to have problems, such as:
  - Version 10.2.0.3.0 of the 10g Release 2 JDBC driver has been noted to produce occurrences of error ORA-01461. The [Oracle Support](#) site has further details on this Oracle server issue, although you will need an Oracle support account to access this site.
  - Version 10.2.0.1.0 of the 10g Release 2 JDBC driver hangs with some databases.
  - The 10g Release 1 JDBC driver (10.1.0.4) does not hang, but [throws `ArrayIndexOutOfBoundsException`](#).

### 4. Configure Your JIRA Server to Connect to Your Oracle Database

There are two ways to configure your JIRA server to connect to your Oracle database:

- [Using the JIRA Configuration Tool](#) (JIRA Standalone distributions only) or
- [Manually](#)

#### Please Note:

- The [JIRA Configuration Tool](#) requires a Java platform to be installed and configured on your operating system. We recommend using a [Java platform supported by JIRA](#) — refer to [JIRA Requirements](#) for details.
- If you use the JIRA WAR distribution or have a console-only connection to your JIRA server, you will need to configure JIRA with Oracle [manually](#).

#### 4.1 Connecting JIRA to Oracle Using the JIRA Configuration Tool

 The [JIRA Configuration Tool](#) is only available with [JIRA Standalone](#).

**To start the JIRA Configuration Tool:**

- **On Windows:**  
Open a command prompt and run `config.bat` in the `bin` subdirectory of the [JIRA Installation Directory](#).
- **On Linux/Unix:**  
Open a console and execute `config.sh` in the `bin` subdirectory of the [JIRA Installation Directory](#).

 **Please Note:** You may need to set the `JAVA_HOME` environment variable to run the JIRA Configuration Tool. See [Installing Java](#) for details.

**To connect JIRA to Oracle using the JIRA Configuration Tool:**

1. Click the '**Database**' tab.
2. From the 'Database type' drop-down choose '**Oracle**'.

 The JIRA Configuration Tool will display your current database configuration settings if any are already set.

The screenshot shows the 'JIRA Configuration Tool' window with the 'Database' tab selected. The 'Database type' is set to 'Oracle'. The 'Connection Details' section contains the following fields: Hostname (dbserver), Port (1521), SID (ORCL), Username (jiradbuser), and Password (password). The 'Connection Pool' section has a 'Pool Size' field set to 15. A 'Test Connection' button is located below the fields, and 'Save' and 'Close' buttons are at the bottom.

3. Fill in the connection details for your Oracle database.
  - 'Hostname' — The name or IP address of the machine that the Oracle server is installed on.
  - 'Port' — The TCP/IP port that the Oracle server is listening on. The default port number for Oracle is '1521'.
  - 'SID' — The Oracle 'System Identifier'. The default value for most Oracle servers is 'ORCL'. If you are using the Oracle Express Edition, this will be 'XE'.
  - 'Username' — The user that JIRA uses to connect to the Oracle server.
  - 'Password' — The user's password to authenticate with the Oracle server.
4. JIRA keeps a pool of database connections open to the database server. You can set the maximum size of this pool in the 'Pool Size' text field.
5. After typing in your settings, click the **'Test Connection'** button to test the connection settings. The tool will attempt to connect to the database, and give a message with the results.
6. Click **'Save'** to save your settings when you are done.
  - i Please Note:**
    - The JIRA Configuration Tool will save your database configuration to a `dbconfig.xml` file in your **JIRA Home Directory**.
    - If you had previously specified any additional custom settings while manually configuring JIRA with Oracle [below](#) (for example, adding the `<connection-properties>SetBigStringTryClob=true</connection-properties>` element to your `dbconfig.xml` file), these custom settings will be deleted upon clicking the **'Save'** button and you will need to reinstate them [manually](#).
    - JIRA must be restarted for your new settings to take effect.

Congratulations — you have finished! Proceed to 'Next Steps' [below](#).

## 4.2 Connecting JIRA to Oracle Manually

1. Edit the `dbconfig.xml` file at the root of your **JIRA Home Directory**.
  - i** If this file does not exist, create the file, copy and paste the example XML code below into this file and edit the pasted XML as required.

**Please Note:**

Ensure that the `<database-type/>` element's content specifies your type of database, as shown below. If you forget to do this and you start JIRA, your database tables may be created incorrectly. Refer to our [Incorrect database type specified](#) documentation if this happens to you.

- w** When editing your `dbconfig.xml` file, escape any `'&'` characters by adding `'amp;'` to the end of each one.

```
<?xml version="1.0" encoding="UTF-8"?>

<jira-database-config>
  <name>defaultDS</name>
  <delegator-name>default</delegator-name>
  <database-type>oracle10g</database-type>
  <schema-name></schema-name>
  <jdbc-datasource>
    <url>jdbc:oracle:thin:@<hostname>:<port number>:<SID></url>
    <driver-class>oracle.jdbc.OracleDriver</driver-class>
    <username>jiradbuser</username>
    <password>[enter db password]</password>
    <pool-size>15</pool-size>
  </jdbc-datasource>
</jira-database-config>
```

 On a default Oracle server installation, your `<url/>` element's content might look similar to — `jdbc:oracle:thin:@dbserver:1521:ORCL`

2. Save your edited `dbconfig.xml` file (at the root of your [JIRA Home Directory](#)).

 JIRA must be restarted for your new settings to take effect.

#### Please Note:

- If you start experiencing problems when dealing with custom workflows or working with issues that have long descriptions, comments or custom field values, try adding the element `<connection-properties>SetBigStringTryClob=true</connection-properties>` as a child of the `</jdbc-datasource>` element in your `dbconfig.xml` file. Adding this connection property may overcome these problems. Be aware that you will need to restart JIRA for this setting to take effect.

## 5. Next steps

You should now have JIRA configured to connect to your Oracle database. The next step is to start it up!

- If you are using JIRA Standalone, start it up and watch the logs for any errors.
- If you are using the JIRA WAR, rebuild and redeploy the webapp in your application server.

## Installation notes

Please see [JIRA and Oracle](#).