

# Install PRIME

This article describes the server side installation of the [Nexus PRIME](#) software package.

See here for more information on installation requirements: [PRIME requirements and interoperability](#).

[Expand/Collapse All](#)

## Prerequisites

### ▼ Prerequisites

The following prerequisites apply:

- See here for more information on installation requirements: [PRIME requirements and interoperability](#).
- A valid license must be available.
- A database must be available, according to the database requirements in [PRIME requirements and interoperability](#).

## Install PRIME

### ▼ Install Java

To install Java on the PRIME server:

1. Download and install Oracle Java JRE (or JDK) according to <https://java.com/>. For information on the supported Java versions, see [PRIME requirements and interoperability](#). Keep the default installation path, since it is needed again for the Tomcat installation.

If you get prompted to remove old Java installations, do not do that until you have installed PRIME with the latest Java update, and know that it works.

2. To set the cryptographic policy of Java, edit the policy settings:
  1. Open the file `<java-home>/lib/security/java.security` for editing.
  2. Uncomment the following line, to allow use of the included policy files:

#### Example: crypto.policy

```
crypto.policy=unlimited
```

3. Save and close the file.

PRIME default settings require use of the unlimited policy files, to allow PRIME to use strong encryption algorithms and key sizes. Due to national regulations in some countries Java is shipped by default with a restricted setup. If the national regulations does not allow to

This article is valid from Nexus PRIME 3.9.

## Related information

- [PRIME requirements and interoperability](#)
- [PRIME installation and upgrade](#)

## Links

- <https://java.com/>
- <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- <http://tomcat.apache.org/>

use this policy, the encryption and signing settings in PRIME need to be changed.

For older Java installations, previous to Java 8 Update 151, the policy files are not included in the default installation. In that case, you must download and install Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction policy files, according to: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

#### ▼ Install Tomcat

Install Apache Tomcat on the PRIME server:

1. Download the installation program from Apache Tomcat: <http://tomcat.apache.org/>.
2. Start the installation by executing the downloaded .exe file.
3. Follow the install wizard:
  1. In **Choose Components**, select installation type **Custom**. Check **Service Startup**. Uncheck **Start Menu Items**, **Documentation**, **Manager**, **Host Manager**, and **Examples**. Click **Next**.
  2. In **Configuration Options**, enter port numbers (required) and a **Windows Service Name**.

You might stick to the Tomcat default ports.

Always take into consideration the customers infrastructure (for example firewall settings and conflicting third party systems using the same port).

For guidelines on choosing ports, the IANA (The Internet Assigned Numbers Authority) is a good resource.

Click **Next**.

3. In Java Virtual Machine, browse for the folder where Java Runtime Environment (JRE) is located.

The location of the JRE folder determines if Tomcat is started as a 32-bit or 64-bit application, and if the Nexus PRIME Server is started and operated in 32-bit or 64-bit mode.

Click **Next**.

4. Enter a **Destination Folder**, for example *C:\Program Files\Wexus-PRIME-Tomcat8*.
5. Click **Install**. The installation is started. Follow the wizard to finish the installation.

#### **IBM Websphere**

IBM Websphere installation is not described in this instruction. To install IBM Websphere, external integration expertise is needed.

## ▼ Configure Tomcat server

To configure the Tomcat server:

1. Open the configuration file `server.xml` in the Tomcat folder, for example:  
`C:\Program Files (x86)\Apache Software Foundation\Tomcat 8.0\conf\server.xml`
2. Add or update the values in bold:

### Example: server.xml

```
<Connector port="8080" protocol="HTTP/1.1"  
connectionTimeout="60000"  
redirectPort="8443"  
maxPostSize="-1" />
```

The `redirectPort="8443"` parameter should be used for an SSL configuration, but has no significance for a non-SSL connection (as in this case).

## ▼ Copy PRIME program files

To copy the PRIME program files:

1. Go to applications and connectors in the PRIME download area, for example:  
**<prime\_version>\10\_Software\20\_Applications\_and\_Connectors**
2. Copy the following .war files into the `webapps` subfolder of the Tomcat installation directory:
  - `prime_designer.war`
  - `prime_explorer.war`
  - `prime_tenant.war`
  - `ussp.war`
3. Go to tools in the PRIME download area, for example:  
**<prime\_version>\10\_Software\20\_Tools**
4. Unzip the file **`db_update.zip`**.
5. Copy the following file to the `db_update` folder of the PRIME installation folder, for example:
  - On Windows: **<prime\_home>\db\_update\create\_db\_with\_default\_records.bat**
  - On Linux: **<prime\_home>\db\_update\create\_db\_with\_default\_records.sh**

## ▼ Start up Tomcat

To configure memory settings and start up Tomcat:

1. Open the Tomcat Properties either via the System Tray, by clicking **Configure** or via **Configure Tomcat** in the **Apache Tomcat** program group in the windows start menu.
2. Go to the **Java** tab.
3. In order to obtain the best possible performance from the Tomcat service, and therefore the Nexus PRIME applications, enter values for the following parameters. Consider the **Notes on memory sizes** below when choosing the values:

Parameter	Example value	Corresponding Java option
Initial Memory pool	512 MB	Xms
Maximum Memory pool	2048 MB	Xmx
Thread stack size	2048 KB	Xss

#### Notes on memory sizes

- The **Initial Memory pool** value can vary significantly depending on the requirements in the projects and the expected load on the Application server. Consider how many background processes are running and how many concurrent users are expected in Explorer, Designer, and USSP, as well as how many sub systems will be connected to PRIME.
- The **Maximum Memory pool** value should be aligned to the available memory on the Server, reserving some memory for OS background processes.
- Experiences in projects show that a good balance between **Xms** and **Xmx** is 1:4. This may also vary according specific project requirements.
- If you use Java 32-Bit, keep in mind that the **Xmx** must be less than 1500 MB.
- If the Tomcat service does not start up, there may not be enough free memory available. In this case, the values must be reduced. Details can also be found in the Tomcat log files.
- Additional configuration settings are required for Tomcat for certain functions. These are, for example:
  - SSL/TLS configuration for HTTPS connections  
For more information, see [Configure https for Tomcat](#).
  - Certificate-based login for Nexus PRIME Designer, Explorer or Nexus USSP  
For more information, see [Set up certificate-based login to Nexus PRIME](#).

4. Go to the **General** tab. Verify the information and set **Startup type** to **Automatic**. Click **Start**.
5. The Tomcat server is started and unzips the .war files into the new folders:
  - *prime\_designer*
  - *prime\_explorer*
  - *prime\_tenant*
  - *ussp*

This process usually takes a few minutes.

6. When the program files have been installed, stop Tomcat again to configure the database connection:  
In the **General** tab, click **Stop**.

#### Initialize database

Nexus PRIME uses a tool called Liquibase to initialize the database. This allows database changes to be made independently of the underlying database type, therefore making it easy to maintain the database schema for different databases with maximum flexibility.

To initialize the database:

1. Go to the `db_update` folder of the PRIME installation folder, for example:  
`<prime_home>\db_update\`

2. Open the following file:
  - On Windows: `create_db_with_default_records.bat`
  - On Linux: `create_db_with_default_records.sh`

**Example: create\_db\_with\_default\_records.bat**

```
rem set DRIVER=org.postgresql.Driver
rem set DB_PREFIX=decode('
rem set DB_POSTFIX=', 'hex')

rem set
DRIVER=com.microsoft.sqlserver.jdbc.SQLServerDriver
rem set DB_PREFIX=0x
rem set DB_POSTFIX=

rem set DRIVER=oracle.jdbc.OracleDriver
rem set DB_PREFIX=hextoraw('
rem set DB_POSTFIX='')

rem set DRIVER=com.ibm.db2.jcc.DB2Driver
rem set DB_PREFIX=
rem set DB_POSTFIX=

set DRIVER=org.h2.Driver
set DB_PREFIX='
set DB_POSTFIX='
```

3. Make updates in the file to reflect your database:
  1. Uncomment the three lines for your database type, by removing `rem` on each line.
  2. Comment out the database types that are not required, by adding `rem` on each line. For example, comment out the H2 database that is selected by default.
  3. Set the database connection parameters: the JDBC URL of the database server, a user with permissions for schema modifications, and password. For available parameters, see [Database properties file](#).

**Example: database connection parameters**

```
set
URL=jdbc:h2: "%-dp0../h2_app/prime";AUTO_SERVER=T
RUE;DB_CLOSE_ON_EXIT=FALSE
set USER=sa
set PASSWORD=
```

For more examples, see [Examples: database.properties files](#).

4. Save the file.

4. Run the batch file (on Windows) or shell script (on Linux).

Note that the computer on which these scripts are run requires access to the database server. If the script is run on the application server, this should not be a problem because the connection is also needed at runtime.

5. After successful execution, the following message is displayed on the console:  
**Liquibase Update Successful**

#### Configure database connection

Nexus PRIME supports several database types from different vendors. The standard PRIME installation includes an H2 database for demo and testing purposes. For a production system, it is recommended to use another database, since the H2 database is not suitable for productive operation.

The database connection is managed via JDBC. The available functionality includes the relevant JDBC database drivers for the databases that are supported by Nexus PRIME.

The database type and the connection data have to be configured in the file `database.properties`. The configuration must be done separately for each PRIME application: Designer, Explorer, and Tenant.

In a production environment, different databases are normally used for Nexus PRIME Designer and Explorer. That means that the different `database.properties` files contain different configurations.

To configure the database type and connection data:

1. For PRIME Designer:
    1. Open this file in a text editor:  
`<PathToTomcat>/webapps/prime_designer/WEB-INF/classes/database.properties`
    2. Configure the parameters for your database for PRIME Designer.  
For more information on the available parameters and examples, see [Database properties file](#).
  2. For PRIME Explorer:
    1. Open this file in a text editor:  
`<PathToTomcat>/webapps/prime_explorer/WEB-INF/classes/database.properties`
    2. Configure the parameters for your database for PRIME Explorer.  
For more information on the available parameters and examples, see [Database properties file](#).
  3. For PRIME Tenant:
    1. Open this file in a text editor:  
`<PathToTomcat>/webapps/prime_tenant/WEB-INF/classes/database.properties`
    2. Configure the parameters for your database for PRIME Tenant.  
For more information on the available parameters and examples, see [Database properties file](#).
  4. Restart Tomcat to establish the database connection.
-

▼ Create start commands to PRIME client applications

To create start commands to the PRIME client applications:

1. To create a link to PRIME Explorer, PRIME Tenant, and User Self-Service Portal, edit the host name and, if needed, the port number in the default URLs to match your environment. The resulting link can be called in a browser.

The default URLs are the following. For Explorer and Tenant, a corresponding file `index.html` is part of each `.WAR` file:

- Explorer:

Default URL to Explorer
<code>http://&lt;PRIMEHOSTNAME&gt;:8080/prime_explorer/</code>

- Designer:

Default URL to Designer
<code>http://&lt;PRIMEHOSTNAME&gt;:8080/prime_designer/</code>

Tenant:

Default URL to Tenant
<code>http://&lt;PRIMEHOSTNAME&gt;:8080/prime_tenant/</code>

- User Self-Service Portal:

Default URL to User Self-Service Portal
<code>http://&lt;PRIMEHOSTNAME&gt;:8080/ussp/</code>

If you want to do chip encoding in the USSP, for example to renew certificates, ensure that the JRE is enabled in the Browser.

where `<PRIMEHOSTNAME>` is the respective server name or the IP address of the server on which the Nexus PRIME application is installed.

2. If no tenant selection dialog should be displayed, add the parameters `tenantId` and `language` in the application URLs:

#### Example: tenantId and language

```
http://<PRIMEHOSTNAME>:8080/prime_explorer/jsp.mainframe.risc?ccstyle=act3_risc&tenantId=<tenantId>&language=<language>
```

where <tenantId> is replaced by for example 1, 2, or 3 and <language> is replaced by en, de, sv, fr, and so on.

3. To create start commands to the Explorer Java client, edit the path to Java, host name, and, if needed, the port number to match your environment. These applications are started directly using Java Web Start, with the associated Java start program *javaws.exe*.

This is the default command to the Explorer Java client:

#### Default start command for Explorer Java client

```
<pathToJava>\javaws.exe  
"http://<PRIMEHOSTNAME>:8080/prime_explorer/start.jnlp"
```

where <pathToJava> is the path to the JRE installation, for example *C:\Program Files\Java\jre1.8.0\_40\bin* and <PRIMEHOSTNAME> is the server name or the IP address of the server on which Nexus Explorer is installed.

#### ▼ Initial configuration

Go on to do [Initial configuration of PRIME](#).

See also [PRIME installation and upgrade](#) for more configuration steps.

## Database properties file

#### ▼ Database properties

Parameter name	Parameter value
----------------	-----------------

dataSourceType	<p>Type or manufacturer of database. Permitted values:</p> <ul style="list-style-type: none"> <li>• H2 - File based H2 database for development and demo purposes. (not suitable for productive operation)</li> <li>• MS - Microsoft SQL Server 2008/2012/2014 Database</li> <li>• ORA - Oracle 10/11/12 Database</li> <li>• PG - PostgreSQL Database</li> <li>• DB2 - IBM DB2 Database</li> </ul>
dataSourceType.jdbcURL	<p>Complete JDBC URL for the respective database. This is different depending on the type of database and has the following basic structure:</p> <ul style="list-style-type: none"> <li>• For H2 database with multi-user access option: jdbc:h2:&lt;Path_to_DB&gt;/&lt;DB_Name&gt;;AUTO_SERVER=TRUE</li> <li>• For MS SQL server with standard instance: jdbc:sqlserver://&lt;DB Hostname_or_IP&gt;:&lt;DB Server_Port&gt;;DatabaseName=&lt;DB_Name&gt;</li> <li>• For MS SQL server with named instance: jdbc:sqlserver://&lt;DB Hostname_or_IP&gt;\&lt;Instance_Name&gt;;DatabaseName=&lt;DB_Name&gt;</li> <li>• For Oracle database: jdbc:oracle:thin:&lt;DB Hostname_or_IP&gt;:&lt;DB Server_Port&gt;:&lt;DB_Name&gt;</li> <li>• For PostgreSQL database: jdbc:postgresql://&lt;DB Hostname_or_IP&gt;:&lt;DB Server_Port&gt;/&lt;DB_Name&gt;</li> <li>• For DB2 database: jdbc:db2://&lt;DB Hostname_or_IP&gt;:&lt;DB Server_Port&gt;/&lt;DB_Name&gt;</li> </ul>

	DB type	Description
	<Path_to_DB>	Relative or absolute folder path to the database (only with H2)
	<DBHostname_or_IP>	Name or IP address of database server
	<Instance_Name>	Instance name of database installation (only with MS SQL server)
	<DBServer_Port>	<p>Port number of database server. The port numbers are usually allocated automatically or manually during the installation of the respective database.</p> <p>The standard port numbers of the respective databases are:</p> <ul style="list-style-type: none"> <li>• for sqlserver: 1433</li> <li>• for oracle: 1521</li> <li>• for postgresql: 5432</li> <li>• for db2: 50000</li> </ul>
	<DB_Name>	Actual name of database
<code>dataSourceType.user</code>		User name for logging into database
<code>dataSourceType.password</code>		Password for logging into database
<code>quartzDriverDelegateClass</code>		Quartz Scheduler needs his own DB connection. Select the JDBC-driver her for Quartz

## Examples: database.properties files

### ▼ H2 example

This example describes the database named `prime`, stored in the folder `h2_application` in the PRIME installation folder.

### H2 example

```
dataSourceType=H2
dataSource.jdbcUrl=jdbc:h2:h2_application/prime;AUTO_SERVER=TRUE
dataSource.user=sa
dataSource.password=

quartzDriverDelegateClass =
org.quartz.impl.jdbcjobstore.StdJDBCDelegate
```

### MS SQL server example: standard instance

#### MS SQL standard instance example

```
dataSourceType=MS
dataSource.jdbcUrl=jdbc:sqlserver://NexusDBServer:1433;DatabaseName=PRIME
dataSource.user=Nexus
dataSource.password=Secret

quartzDriverDelegateClass =
org.quartz.impl.jdbcjobstore.MSSQLDelegate
```

### MS SQL server example: named instance

#### MS SQL named instance example

```
dataSourceType=MS
dataSource.jdbcUrl=jdbc:sqlserver//NexusDBServer\ACCESS;DatabaseName=PRIME
dataSource.user=Nexus
dataSource.password=Secret

quartzDriverDelegateClass =
org.quartz.impl.jdbcjobstore.MSSQLDelegate
```

### Oracle example

#### Oracle example

```
dataSourceType=ORA
dataSource.jdbcUrl=jdbc:oracle:thin:@//NexusDBServer:1521/PRIME
dataSource.user=Nexus
dataSource.password=Secret

quartzDriverDelegateClass =
org.quartz.impl.jdbcjobstore.oracle.OracleDelegate
```

#### ▼ PostgreSQL example

#### PostgreSQL example

```
dataSourceType=PG
dataSource.jdbcUrl=jdbc:postgresql://NexusDBServer:5432/PRIME
dataSource.user=Nexus
dataSource.password=Secret

quartzDriverDelegateClass =
org.quartz.impl.jdbcjobstore.PostgreSQLDelegate
```

#### ▼ DB2 example

#### DB2 example

```
dataSourceType=DB2
dataSource.jdbcUrl=jdbc:db2://NexusDBServer:50000/PRIME
dataSource.user=Nexus
dataSource.password=Secret

quartzDriverDelegateClass =
org.quartz.impl.jdbcjobstore.DB2v8Delegate
```