

Sakai 19 Install Guide (Binary)

1.0 Sakai 12 pre-requisites

To begin with the following items must be installed (you may skip any you all ready have installed).

1. Java 1.8
2. Git
3. Maven 3.0 or later
4. Tomcat 8

Sakai 11.0 was QA tested with jre-1.8.0_77, maven-3.2.3, tomcat-8.0.32

2.0 Get the binary archive

⚠ Binary limitations. Post-release patches are not always packaged as jars, so a binary install forces a relatively static implementation that cannot be as readily patched later.

The binary distribution of Sakai provides a shortcut for those that already have Tomcat in place and configured as needed (including the database setup and configuration) and it does so by providing a pre-built Sakai that can simply be dropped into place. All you need to do is unpack the binary archive at the root Tomcat directory and the appropriate *.wars, *.jars, etc., for the Sakai application will be deposited in the correct locations.

The latest Sakai binary archive is available at <http://source.sakaiproject.org/release/>.

3.0 Verify/Install Java 1.8

Oracle's Sun [Java SE 8](#), a.k.a Java 1.8, is the required version to use with the Sakai. Certain files, such as *.jsp and *.jws, require compilation so downloading and attempting to use only the run time environment (JRE 8.0) will not suffice.

To confirm that Java is both installed on your system and is the correct version for Sakai, run `java -version` from the command line:

```
java -version
```

If Java is installed, basic version and build information will be displayed. Example output:

```
java version "1.8.0_111"  
Java(TM) SE Runtime Environment (build 1.8.0_31-b13)  
Java HotSpot(TM) 64-Bit Server VM (build 25.31-b07, mixed mode)
```

3.1 Set Java environment variables

Several environment variables and related properties must be set for Java. For UNIX operating systems one typically modifies a startup file like `~/ .bash_login` to set and export shell variables while Mac users typically set and export environment variables in `. bash_profile`. For Windows, go to Start -> Control Panel -> System -> Advanced -> Environment Variables and set `JAVA_HOME` via the GUI.

Set the `JAVA_HOME` environment variable to point to the base directory of your Java installation and add Java's `/bin` directory to the `PATH` environment variable.

i If the variable `JRE_HOME` is already set or if you want to use a particular JRE if you have more than one JRE installed on your machine then you'll want to set the `JRE_HOME` variable as well. `JRE_HOME` is what Apache Tomcat uses when it starts up, but it defaults to use `JAVA_HOME` if `JRE_HOME` is not set. In most cases, setting `JAVA_HOME` should cover both cases sufficiently.

Variable	Unix	Mac	Windows
<code>JAVA_HOME</code>	<code>export JAVA_HOME=/usr/java/java-current</code>	<code>export JAVA_HOME=/Library/Java/Home</code>	<code>JAVA_HOME=C:\jdk1.8.0_31</code>
<code>PATH</code>	<code>export PATH=\$PATH:\$JAVA_HOME/bin</code>	<code>export PATH=\$PATH:\$JAVA_HOME/bin</code>	<code>;C:\jdk1.8.0_31\bin</code>

⚠ Windows: append the string to the end of the `Path` system variable

Set `JAVA_OPTS`

The default Java virtual machine (JVM) settings are insufficient for an application of Sakai's size. As a result several JVM parameters must be increased for Sakai to run, while others may need to be adjusted for optimal performance.

✔ We recommend that you define these settings in Tomcat's `/bin` directory in a file named `setenv.sh` (Unix/Mac) or `setenv.bat` (Windows). See the "Install Tomcat 8" section below for more details.

Specify a Language and Locale (optional)

You can define the default language/locale when starting Sakai by setting the system properties `-Duser.language` and `-Duser.region`. For information on supported languages see the release notes or visit the [i18N Work Group space](#).

```
-Duser.language=pt
-Duser.region=PT
```

In the case your locale is not fully supported in Java (as it happens with Basque or Mongolian languages) you should read this information:

[Endorsed I18n Project](#)

Specify an HTTP Proxy (optional)

In environments where local network policy or firewalls require use of an upstream HTTP proxy/cache, Sakai needs to be configured accordingly. Otherwise components or services which use HTTP requests, such as the `BasicNewsService` for RSS feeds in the News tool, cannot retrieve data from the target URLs. This can be fixed with the following `JAVA_OPTS` arguments:

```
-Dhttp.proxyHost=cache.some.domain
-Dhttp.proxyPort=8080
```

4.0 Install Tomcat 8

- Apache Tomcat - <http://tomcat.apache.org/>
- **Note:** Always do a fresh install of Tomcat 8
- **Note:** For Sakai versions prior to 11.4 please use Tomcat 8.0.34 or earlier. Later versions of Tomcat may cause problems.
example:  [SAK-31608](#) - Sakai has startup warnings on Tomcat 8.0.39+ 
- **Note:** Windows users should ensure that there are no spaces in the complete tomcat path as this causes errors with JSF tools in Sakai
GOOD: `C:\opt\tomcat\`, `C:\sakaistuff\installs\tomcat\`
BAD: `C:\program files\tomcat\`, `C:\opt\apache tomcat 8.0.43\`

1. Download Tomcat 8 - <https://tomcat.apache.org/download-80.cgi>

- Windows users should get the zip file instead of installing a service
It makes viewing the tomcat logs easier and it is easier to configure

2. Extract to `/opt` (symlink the `apache-tomcat-8.0.x` directory to `tomcat` after extracting)

- Example (assuming you have saved the file as `/opt/apache-tomcat-8.0.x.tar.gz`)

```
cd /opt
tar xzvf apache-tomcat-8.0.x.tar.gz
ln -nsf apache-tomcat-8.0.x tomcat
```

- Windows users should either rename the directory or, if comfortable, create a directory junction using an elevated cmd prompt:

```
mklink /J C:\apache-tomcat-8.0.x C:\tomcat
```

3. Modify `conf/server.xml` for international character support

- a. Add `URIEncoding="UTF-8"` to the Connector element
 - `<Connector port="8080" URIEncoding="UTF-8" ...`

4. Set environment variable: `CATALINA_HOME=/opt/tomcat`

5. Add `$CATALINA_HOME/bin` to `PATH`

6. Setup the `SETENV` file in the `tomcat/bin` directory with `JAVA_OPTS` (from [Install Java 1.8](#)).

✔ Populate Database With Demo Data

Instead of starting with an empty database you can add the flag `-Dsakai.demo=true` in addition to the others shown here. During your first Tomcat launch it will populate your database with test data (students, courses, etc). Once the data has been created remove this flag for subsequent Tomcat launches. Be advised that this flag will not trigger on a database with data already in it so you must make this choice during the initial setup. It should also be noted that without this flag one will not be able to create any course sites in Sakai unless Academic Term, Subject, Course, and Section data is manually added to the "cm_" tables in the database.



We recommend using `-Djava.util.Arrays.useLegacyMergeSort=true` until this issue can be resolved -

[SAK-31707](#) - Broken Equality Comparators in Gradebook RESOLVED

- a. Mac/Linux: Create a file called **setenv.sh** with the following (alternately, you can put this into your **.bashrc** file so they're automatically executed):

Tomcat 8 Mac/Linux

```
export JAVA_OPTS="-server -d64 -Xms1g -Xmx2g -Djava.awt.headless=true -XX:+UseCompressedOops -XX:
+UseConcMarkSweepGC -XX:+DisableExplicitGC"
JAVA_OPTS="$JAVA_OPTS -Dhttp.agent=Sakai"
JAVA_OPTS="$JAVA_OPTS -Dorg.apache.jasper.compiler.Parser.STRICT_QUOTE_ESCAPING=false"
JAVA_OPTS="$JAVA_OPTS -Dsakai.security=$CATALINA_HOME/sakai/"
JAVA_OPTS="$JAVA_OPTS -Duser.timezone=US/Eastern"
JAVA_OPTS="$JAVA_OPTS -Dsakai.cookieName=SAKAI2SESSIONID"
JAVA_OPTS="$JAVA_OPTS -Dcom.sun.management.jmxremote -Dcom.sun.management.jmxremote.port=8089 -
Dcom.sun.management.jmxremote.local.only=false -Dcom.sun.management.jmxremote.authenticate=false -
Dcom.sun.management.jmxremote.ssl=false"
```

- b. Windows(PC): Create a file called **setenv.bat** with the following:

Tomcat 8 Windows

```
set JAVA_OPTS=-server -Xmx1028m -XX:MaxMetaspaceSize=512m -Dorg.apache.jasper.compiler.Parser.
STRICT_QUOTE_ESCAPING=false -Djava.awt.headless=true -Dcom.sun.management.jmxremote -Dhttp.
agent=Sakai -Djava.util.Arrays.useLegacyMergeSort=true -Dfile.encoding=UTF8
```

7. [OPTIONAL] Delete the default webapps from the webapps dir

```
rm -rf webapps/*
```

Improve startup speed

You can improve startup speed under Tomcat 8 significantly.

For Sakai 12 and earlier: Edit the file `conf/context.xml` and add this `JarScanFilter` block to the `<Context>`

```
<JarScanner>
  <!-- This is to speedup startup so that tomcat doesn't scan as much -->
  <JarScanFilter defaultPluggabilityScan="false"
defaultTldScan="false"
tldScan="jsf-impl-*.jar,jsf-widgets-*.jar,myfaces-impl-*.jar,pluto-taglib-*.jar,sakai-sections-app-util-*.jar,
spring-webmvc-*.jar,standard-*.jar,tomahawk*.jar,tomahawk-*.jar"/>
</JarScanner>
```

For Sakai 19 and later: After the JSF 2.3 migration, users may update the tomcat configuration to this one:

```
<JarScanner>
<!-- This is to speedup startup so that tomcat doesn't scan as much -->
<JarScanFilter defaultPluggabilityScan="false" />
</JarScanner>
```

5.0 Unpack the Sakai binary distribution

The Sakai binary archive is available at <http://source.sakaiproject.org/release/>.

Download the archive into `$CATALINA_HOME` and then unpack the archive. Then start up Tomcat by running the Tomcat start up script in `$CATALINA_HOME/bin`. Unix/Mac users should make sure that they have write permissions to the Tomcat servlet container files and directories before proceeding or start up permission errors may occur.

⚠ If you wish to configure a binary installation you'll need to manually create a `/sakai` folder in `$CATALINA_HOME` to hold your `sakai.properties` file.

5.1 Configure Sakai

The `sakai.properties` file is a central configuration file that is typically stored in a `/sakai` subdirectory relative to the Tomcat home directory (`$CATALINA_HOME`). It is a non-XML text file containing a series of key/value pairs that is read using the load method of [java.util.properties](#). Settings in `sakai.properties` govern everything from setting your institution's name to configuring your database. All settings in `sakai.properties` are read on startup; any changes you make subsequently will only take effect when you restart web application server.

For a source installation the default `default.sakai.properties` file is located in the `config` module:

```
sakai-src/config/configuration/bundles/src/bundle/org/sakaiproject/config/bundle/default.sakai.properties
```

⚠ The bin package does not include a `sakai.properties` file. This is a deliberate exclusion; it eliminates the possibility of overwriting a local `sakai.properties` file if a bin package is opened over an existing Sakai installation.

If you need to override the default settings you must create your own `sakai.properties` file either from scratch or from a known working copy adding new key/value settings in order to customize your installation. We recommend that you review the `default.sakai.properties` file included in the source installation or in the appropriate maintenance branch.

Version	default
10.0	default.sakai.properties
2.9	default.sakai.properties

The default location for your local `sakai.properties` file is `$CATALINA_HOME/sakai`. This folder is not created by Maven during the build and deployment process, so you will have to create it manually or via a script. You can also store Sakai's configuration files outside of your web application server's file hierarchy. For example, in a development environment you may find yourself frequently reinstalling Tomcat and unless you create a build script to automate the Tomcat installation and configuration process avoiding having to recreate `$CATALINA_HOME/sakai` and `sakai.properties` each time has its advantages.

To locate your properties file outside of your web application server environment modify the Java startup command or the `JAVA_OPTS` environment variable and set a system property named `sakai.home`. Make sure your external location is readable and writable by your web application server.

```
-Dsakai.home=/path/to/desired/sakai/home/
```

5.2 Configure Database

[Sakai 11 database support details](#)

Make sure to include a MySQL connector jar in your `CATALINA_HOME/lib`.

Create a Sakai database with a default character set of UTF-8. Create a user account to which you will assign all permissions.

Use any database name and username you wish and then update your `sakai.properties` file (or even better, create a `local.properties` file, which has precedence).

In the example below the database is named "sakaidatabase" and the user account is "sakaiuser".

Excerpt from sakai.properties

```
username@javax.sql.DataSource=sakaiuser  
password@javax.sql.DataSource=sakaipassword
```

MySQL settings

```
vendor@org.sakaiproject.db.api.SqlService=mysql  
driverClassName@javax.sql.DataSource=com.mysql.jdbc.Driver  
hibernate.dialect=org.hibernate.dialect.MySQL5InnoDBDialect  
url@javax.sql.DataSource=jdbc:mysql://127.0.0.1:3306/sakaidatabase?useUnicode=true&characterEncoding=UTF-8  
validationQuery@javax.sql.DataSource=select 1 from DUAL  
defaultTransactionIsolationString@javax.sql.DataSource=TRANSACTION_READ_COMMITTED
```

```
mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 272  
Server version: 5.1.46 MySQL Community Server (GPL)  
  
Copyright (c) 2000, 2010, Oracle and/or its affiliates. All rights reserved.  
This software comes with ABSOLUTELY NO WARRANTY. This is free software,  
and you are welcome to modify and redistribute it under the GPL v2 license  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
mysql> create database sakaidatabase default character set utf8;  
Query OK, 1 row affected (0.00 sec)  
  
mysql> grant all on sakaidatabase.* to sakaiuser@'localhost' identified by 'sakaipassword';  
Query OK, 0 rows affected (0.00 sec)  
  
mysql> grant all on sakaidatabase.* to sakaiuser@'127.0.0.1' identified by 'sakaipassword';  
Query OK, 0 rows affected (0.00 sec)  
  
mysql> flush privileges;  
Query OK, 0 rows affected (0.00 sec)  
  
mysql> quit
```

6.0 Start/Stop Tomcat

Start/stop Tomcat from the terminal by running the appropriate startup/shutdown script located in \$CATALINA_HOME/bin:

Unix/Mac

```
sh startup.sh  
sh shutdown.sh
```

Windows

```
startup.bat  
shutdown.bat
```