

Sakai 11 upgrade information

Upgrading from an earlier version of Sakai typically involves a database conversion (for which scripts are supplied with the release), an update to custom skins and possibly changes to any custom code.

1.0 Converting Favorited Sites

Sakai 11 changes the way that sites marked as "favorites" are stored in the database. If you are upgrading from an earlier version of Sakai, you should run the "ConvertUserFavoriteSitesSakai11" script to migrate your users' existing favorite sites to the new format.

You can run this script at any time following the upgrade to Sakai 11, and Sakai does not need to be running for the script to work. If you are planning an upgrade, a good time to run this is right before starting Sakai 11 up for the first time.

The time required for the conversion will depend on how many users you have in your system. As a rule of thumb, you can expect it to take between 2-5 minutes per hundred thousand users.

To run the script, open a shell and run the following commands:

```
cd /path/to/your/tomcat-directory
```

Then, for Unix systems:

```
java -cp "lib/*" -Dtomcat.dir="$PWD" org.sakaiproject.user.util.ConvertUserFavoriteSitesSakai11
```

Or under Windows:

```
java -cp "lib\*" -Dtomcat.dir=%cd% org.sakaiproject.user.util.ConvertUserFavoriteSitesSakai11
```

2.0 Database Conversion Scripts

Please run ALL the scripts from your original version to the version to which you are upgrading. There are two sets of database upgrade scripts, one for MySQL and one for Oracle. Use the appropriate scripts for your database type.

<https://github.com/sakaiproject/sakai-reference/tree/master/docs/conversion>

3.0 Other Database Conversions

Sakai also includes data and schema conversions outside the standard scripts to address data integrity issues for assignments, content hosting, email archive and migration to GradebookNG (the new default Gradebook. Gradebook has been renamed Gradebook Classic as of Sakai 11). Below is a quick summary and overview of the process for each tool.

3.1 MailArchive conversion (2.5 -> 2.6)

[SAK-13584](#) - Further Improve the Performance of the Email Archive and Message API CLOSED ,

[SAK-16554](#) - Email archive conversion script is difficult to use and requires a maven source repository CLOSED : Sakai 2.6

improvements to the Message API require that implementers upgrading from a pre-2.6 version of Sakai run both the 2.6.0 conversion scripts and a second `mailarchive-runconversion.sh` script that can be found in the mailarchive module in order to update your existing mail archive data to take advantage of the 2.6 Email Archive performance improvements.

```
-- SAK-13584 Further Improve the Performance of the Email Archive and Message API. Note you have to run a bash
conversion script on your old mail archive data for it to
-- appear in the new mail archive. The script is in the source as mailarchive-runconversion.sh. Please see the
SAK for more information on this script or SAK-16554 for
-- updates to this script.
```

```
CREATE INDEX IE_MAILARC_SUBJECT ON MAILARCHIVE_MESSAGE
(
    SUBJECT          ASC
);
ALTER TABLE MAILARCHIVE_MESSAGE ADD COLUMN (
    SUBJECT          VARCHAR (255) NULL,
    BODY             LONGTEXT NULL
);
```

3.2 Assignments Tool conversion

⚠ This conversion was a part of the post-2.4 assignments branch so those migrating from a version already running this can disregard this step.

🔴 **SAK-11821** - AssignmentService allows creation of duplicate submission objects, need to separate submitter_id out to form unique constraint, and improve tool performance by using new db columns CLOSED

: The assignment service previously permitted the creation of duplicate submission objects (i.e. two or more submissions for the same student and assignment). While the UI should prevent this from happening, at various stages in the evolution of the Assignments code, bugs, race conditions or other failures have led to duplicate objects being created. This conversion seeks to reconcile those duplicates and add database constraints to prevent this in the future.

In summary, the conversion script is necessary to

1. Remove existing duplicate submission records, if any
2. Prevent future submission duplicates by applying unique constraint on the ASSIGNMENT_SUBMISSION table
3. Improve performance of the Assignment tool

The conversion script does the following to the existing ASSIGNMENT_SUBMISSION table in Sakai database:

1. read in all tuples as AssignmentSubmission object, parse out data such as submitter_id, submit_time, submitted, and graded, and stores those attributes as separate columns in the ASSIGNMENT_SUBMISSION table;
2. Runs through the table, combine and remove submission duplicates (tuples with same "context" and "submitter_id" combination);
3. apply the unique constraint of "context" + "submitter_id" to ASSIGNMENT_SUBMISSION table.

There is a README file with detailed instructions on this process at https://source.sakaiproject.org/svn/assignment/tags/sakai_2-7-0.

3.3 Content Hosting

Migrating from 2.4 to 2.5 requires adding several columns to the database tables used by the Content Hosting Service (the service used by the Resources tool, Dropbox, filepicker and WebDAV). The conversion scripts contain DDL statements to accomplish those changes. You need to run these conversions scripts (or perform the equivalent operations manually) and then convert your existing Resources via one of the methods outlined below, before you will gain the performance improvements Sakai 2.5 offers.

The new columns added to the database tables support a switch from XML serialization to "binary-entity" serialization, and enable Resources to perform faster and use less memory. One of the key areas this impacts is improving the performance of quota calculations.

There are two methods for converting existing Resources, with the first being the recommended option, as it enables all performance improvements when completed:

1. Run the conversion utility, which can be run on a live system. (See [readme](#) for more details.)
 - ⚠ Systems running oracle should read the email threads copied to the comments section below.
2. Let the code convert each Resource as it is accessed.
 - ⚠ This is only recommended for implementations with small datasets, such as pilot deployments; otherwise you should use the above conversion utility.
 - ⚠ While some performance benefits from the binary-entity serialization can be realized immediately using this method, others, such as the quota calculation improvements will not be available until all Resources have been accessed and converted.

Based on the state of the data in the Content Hosting Service tables, it will start up in one of two modes:

1. Binary only - If the code detects on start-up that all of the XML fields are null – as would be the case after running the conversion utility – it will run in binary mode. This means the system will only read and write using binary-entity serialization, and you will be able to fully realize all the performance enhancements that it offers.
2. Dual mode - If the code detects there is still data in the XML fields – as would be the case if the conversion utility has not been run – it will run in dual mode. This means the system will be capable of reading both XML-serialized and binary-entity-serialized resources, but will write using only binary, and will convert any XML data it encounters into binary data. This gives you some of the performance benefits of binary-entity serialization without running the conversion utility, but you will never get the quota-calculation improvements unless all Resources end up converted.

3.4 Using the Import from Site Feature with Gradebook from a pre-Sakai 11 installation upgrading to Sakai 11

Please read this blog post by a senior Sakai developer.

<https://steveswinsburg.wordpress.com/2016/08/26/migrating-to-gradebookng/>

4.0 Portfolios (OSP) permission settings

OSP was turned off by default in Sakai 10 and removed entirely in Sakai 11.

 [SAK-13205](#) - Permissions for CIG Participant are empty for 2.5 upgrade CLOSED : if you are upgrading from a pre-2.5.0 version of Sakai, then you need to force conversion of the OSP permissions by setting `osp.upgrade25=true` in your `sakai.properties` file. (See also [discussion](#).)

5.0 Browser Compatibility

Sakai 11 is designed to work with modern browsers. Recommended browsers (those with the most QA testing effort against them) are Google Chrome and Mozilla Firefox. Internet Explorer 11 (IE 11) and later Microsoft browser options should work fine. If you are upgrading from an earlier version of Sakai (pre-Sakai 11) please make sure that you don't have the property `X-UA-Compatible:IE` set, or it may make Microsoft IE 11 behave as though it is an earlier browser version and therefore not work.

Some iPad and mobile testing has been done, but more work is needed in this area (this is the first release of Sakai using responsive design).

6.0 Upgrading the Sakai skin

<https://github.com/sakaiproject/sakai/tree/master/reference>