

Sakai 1.5 -> 2.0.0 Database Conversion

June 16, 2005

Please direct questions or comments about this document to:

Glenn R. Golden, ggolden@umich.edu

Converting existing Sakai 1.5 based installations to 2.0 is a simple process, but one that requires some local customization. The conversion tasks fall into these categories:

- rename tables and sequences
- drop unused tables and sequences
- create new tables, sequences, indexes
- expand some field lengths
- update site definitions with new tool ids
- update site types
- update site skins
- optionally replace use myWorkspace definitions
- optionally replace the Gateway site

All the conversion can be done with SQL commands against the database.

The conversion to create new tables can also be done by starting an instance of Sakai, with auto-ddl enabled, using the otherwise converted database, and letting it auto-create and populate the new tables. Once this is done, your DBA might want to move things around to different table spaces, etc.

The alternate to running the auto-ddl is to manually find the .sql files that create and populate the new tables from the source code, let your DBA have at them, and run them manually.

The local changes to the conversion process involve picking which optional parts to run, what you want to do about skins and user myWorkspaces, and any additional changes you want to make.

The examples in this document have been written for and tested with Oracle. MySQL syntax may be slightly different.

The SQL command part of the conversion takes just a few minutes.

The commands from this document are in the source code, but not in the released 2.0.0 source code. You can find them here using a browser, svn client, or even webdav:

https://source.sakaiproject.org/svn/trunk/sakai/docs/sakai_1_5-2_0_conversion.sql

Table Renames

This is pretty simple:

```
alter table CHEF_EVENT rename to SAKAI_EVENT;
alter table CHEF_DIGEST rename to SAKAI_DIGEST;
alter table CHEF_NOTIFICATION rename to SAKAI_NOTIFICATION;
alter table CHEF_PREFERENCES rename to SAKAI_PREFERENCES;
alter table CHEF_PRESENCE rename to SAKAI_PRESENCE;
alter table CHEF_SESSION rename to SAKAI_SESSION;
alter table SYLLABUS_DATA_T rename to SAKAI_SYLLABUS_DATA;
alter table SYLLABUS_T rename to SAKAI_SYLLABUS_ITEM;
```

The last two (for the Syllabus app) will fail if you don't already run Syllabus.

There's a sequence that we also need to rename:

```
rename CHEF_EVENT_SEQ to SAKAI_EVENT_SEQ;
```

Drop Unused Tables and Sequences

There is one sequence we no longer use:

```
drop sequence CHEF_ID_SEQ;
```

Expanded Fields

With our new UUID generator, the fields in the database that held ids need to be expanded to hold 36 characters:

```
alter table ANNOUNCEMENT_MESSAGE modify MESSAGE_ID VARCHAR2(36);
alter table CALENDAR_EVENT modify EVENT_ID VARCHAR2(36);
alter table CHAT_MESSAGE modify MESSAGE_ID VARCHAR2(36);
alter table DISCUSSION_MESSAGE modify MESSAGE_ID VARCHAR2(36);
alter table DISCUSSION_MESSAGE modify REPLY VARCHAR2(36);
alter table MAILARCHIVE_MESSAGE modify MESSAGE_ID VARCHAR2(36);
alter table SAKAI_EVENT modify SESSION_ID VARCHAR2(36);
alter table SAKAI_PRESENCE modify SESSION_ID VARCHAR2(36);
alter table SAKAI_SESSION modify SESSION_ID VARCHAR2(36);
alter table SAKAI_LOCKS modify USAGE_SESSION_ID VARCHAR2(36);
```

Tool Id Changes

Tool ids were changes to be consistent. These are what are stored in Site definitions, so we must update the sites:

```

91 update SAKAI_SITE_TOOL
92 set REGISTRATION=concat('sakai', substr(REGISTRATION,5)) where
93 UPPER(substr(REGISTRATION,1,4)) = 'CHEF';
94 update SAKAI_SITE_TOOL set REGISTRATION='sakai.preferences' where
95 REGISTRATION='sakai.noti.prefs';
96 update SAKAI_SITE_TOOL set REGISTRATION='sakai.online' where
97 REGISTRATION='sakai.presence';
98 update SAKAI_SITE_TOOL set REGISTRATION='sakai.siteinfo' where
99 REGISTRATION='sakai.siteinfogeneric';
100 update SAKAI_SITE_TOOL set REGISTRATION='sakai.sitesetup' where
101 REGISTRATION='sakai.sitesetupgeneric';
102 update SAKAI_SITE_TOOL set REGISTRATION='sakai.discussion' where
103 REGISTRATION='sakai.threadeddiscussion';

```

If you have been using the University of Michigan GradTools, you need to do this, also:

```

107 update SAKAI_SITE_TOOL set REGISTRATION='ctools.dissertation' where
108 REGISTRATION='sakai.dissertation';
109 update SAKAI_SITE_TOOL set REGISTRATION='ctools.dissertation.upload'
110 where REGISTRATION='sakai.dissertation.upload';
111 update SAKAI_SITE_TOOL set REGISTRATION='ctools.gradToolsHelp' where
112 REGISTRATION='sakai.gradToolsHelp';

```

Site Types

There may be old site types in your data. This will update them:

```

118 update sakai_site set type='project' where type='CTNG-project';
119 update sakai_site set type='course' where type='CTNG-course';

```

[OPTIONAL] Site Skins

The skin model has changed in Sakai 2. Sites with skin settings that don't exist in your system will look very strange until you fix their skin. You might want to clear out all skins, reverting all sites to the default skin.

```

127 update SAKAI_SITE set SKIN=null;

```

At the University of Michigan, we ran code that detected the presence of the old skin scheme, and updated the skin and the site icon to match the new skin scheme. Here's an example:

```

133 update SAKAI_SITE set SKIN=null, ICON_URL='/ctlib/icon/art.gif' where
134 skin='art.css';
135 update SAKAI_SITE set SKIN=null, ICON_URL='/ctlib/icon/bus.gif' where
136 skin='bus.css';
137 update SAKAI_SITE set SKIN=null where skin='chef.css';

```

For the first two, we detect the old skin name, and clear the skin but set an icon. In the third case, we just clear that skin, as that is not used any more in our system.

You need to design the SQL commands to modify your site definitions to update your skins.

[OPTIONAL] User MyWorkspace Definitions

There are new features, such as Profile, that are available in Sakai that you might want to add to your user's MyWorkspace sites.

One way to do this is to modify the template that is used to create new MyWorkspace sites, and then delete all the existing site definitions. When your users log in, they will get a newly created MyWorkspace.

Deleting the sites can be accomplished with this:

```
delete from sakai_site_user where site_id like '~%' and site_id !=
 '~admin';
delete from sakai_site_tool_property where site_id like '~%' and
 site_id != '~admin';
delete from sakai_site_tool where site_id like '~%' and site_id !=
 '~admin';
delete from sakai_site_page where site_id like '~%' and site_id !=
 '~admin';
delete from sakai_site where site_id like '~%' and site_id != '~admin';
```

This removes all user sites (sites starting with “~”) except the admin site, from all site related tables.

Any data they might have in their workspace (such as in schedule or resources) will be preserved, but modifications they may have made to their site structure will be lost.

Another option is to do nothing, and let the users modify their sites to pick up new features as they wish.

A third option is to craft SQL that inserts site pages and site tools into the sites. This should be possible, and is an exercise left to the brave.

[OPTIONAL] Replace The Gateway Site

You can use this code to install the newly distributed Gateway site. You can also modify the Gateway site (“!gateway”) using the normal tools in Sakai (admin site editor, options from the site itself). Or, if you are happy with your Gateway site, you can leave it alone!

```
delete from sakai_site_user where site_id = '!gateway';
```

```

184 delete from sakai_site_tool_property where site_id = '!gateway';
185 delete from sakai_site_tool where site_id = '!gateway';
186 delete from sakai_site_page where site_id = '!gateway';
187 delete from sakai_site where site_id like '!gateway';
188 INSERT INTO SAKAI_SITE VALUES('!gateway', 'Gateway', null, null, 'The
189 Gateway Site', null, null, null, 1, 0, 0, '', null, null, null, null,
190 1, 0 );
191 UPDATE SAKAI_SITE SET MODIFIEDBY='admin' WHERE SITE_ID = '!gateway';
192 UPDATE SAKAI_SITE SET
193 MODIFIEDON=TO_TIMESTAMP('20031126034522061','YYYYMMDDHHMISSFF3') WHERE
194 SITE_ID = '!gateway';
195 INSERT INTO SAKAI_SITE_PAGE VALUES('!gateway-100', '!gateway',
196 'Welcome', '0', 1 );
197 INSERT INTO SAKAI_SITE_TOOL VALUES('!gateway-110', '!gateway-100',
198 '!gateway', 'sakai.motd', 1, 'Message of the day', NULL );
199 INSERT INTO SAKAI_SITE_TOOL VALUES('!gateway-120', '!gateway-100',
200 '!gateway', 'sakai.iframe', 2, 'Welcome!', NULL );
201 INSERT INTO SAKAI_SITE_TOOL_PROPERTY VALUES('!gateway', '!gateway-120',
202 'special', 'site' );
203 INSERT INTO SAKAI_SITE_PAGE VALUES('!gateway-200', '!gateway', 'About',
204 '0', 2 );
205 INSERT INTO SAKAI_SITE_TOOL VALUES('!gateway-210', '!gateway-200',
206 '!gateway', 'sakai.iframe', 1, 'About', NULL );
207 INSERT INTO SAKAI_SITE_TOOL_PROPERTY VALUES('!gateway', '!gateway-210',
208 'height', '500px' );
209 INSERT INTO SAKAI_SITE_TOOL_PROPERTY VALUES('!gateway', '!gateway-210',
210 'source', '/library/content/gateway/about.html' );
211 INSERT INTO SAKAI_SITE_PAGE VALUES('!gateway-300', '!gateway',
212 'Features', '0', 3 );
213 INSERT INTO SAKAI_SITE_TOOL VALUES('!gateway-310', '!gateway-300',
214 '!gateway', 'sakai.iframe', 1, 'Features', NULL );
215 INSERT INTO SAKAI_SITE_TOOL_PROPERTY VALUES('!gateway', '!gateway-310',
216 'height', '500px' );
217 INSERT INTO SAKAI_SITE_TOOL_PROPERTY VALUES('!gateway', '!gateway-310',
218 'source', '/library/content/gateway/features.html' );
219 INSERT INTO SAKAI_SITE_PAGE VALUES('!gateway-400', '!gateway', 'Sites',
220 '0', 4 );
221 INSERT INTO SAKAI_SITE_TOOL VALUES('!gateway-410', '!gateway-400',
222 '!gateway', 'sakai.sitebrowser', 1, 'Sites', NULL );
223 INSERT INTO SAKAI_SITE_PAGE VALUES('!gateway-500', '!gateway',
224 'Training', '0', 5 );
225 INSERT INTO SAKAI_SITE_TOOL VALUES('!gateway-510', '!gateway-500',
226 '!gateway', 'sakai.iframe', 1, 'Training', NULL );
227 INSERT INTO SAKAI_SITE_TOOL_PROPERTY VALUES('!gateway', '!gateway-510',
228 'height', '500px' );
229 INSERT INTO SAKAI_SITE_TOOL_PROPERTY VALUES('!gateway', '!gateway-510',
230 'source', '/library/content/gateway/training.html' );
231 INSERT INTO SAKAI_SITE_PAGE VALUES('!gateway-600', '!gateway', 'New
232 Account', '0', 6 );
233 INSERT INTO SAKAI_SITE_TOOL VALUES('!gateway-610', '!gateway-600',
234 '!gateway', 'sakai.createuser', 1, 'New Account', NULL );
235

```

236 Getting new Table Created and Populated

237

238 Once your database is converted with the steps you have selected and customized so far,
239 you still need to have the new Sakai applications' database tables created and populated.

240
241 The easiest way to do this is to run an instance of Sakai with auto-ddl turned on. The
242 auto-ddl feature is by default turned on, and can be controlled in your
243 placeholder.properties file.

244
245 The auto-ddl feature will recognize that most of the tables you need are already defined,
246 and leave them alone. But for the new applications, Samigo, Gradebook, Profile, etc,
247 tables will be created and seed data will be inserted.

248
249 Once the Sakai instance starts up, you can shut it down and let your DBA study your
250 resulting database. She might want to move things into different tablespaces, and do
251 other things that we pay them the big bucks to take care of.

252
253 If, on the other hand, said DBA wants to modify the SQL command before running them,
254 then you need to harvest the auto-ddl scripts from the source code. They can be found
255 with a little digging.

256
257 Each application has some source code project that has a src/sql folder. In there you will
258 find a folder for each database technology (hsqldb, mysql, oracle). And in there you will
259 find the .sql file to create and populate the tables for that application.

260
261 For example, for the Syllabus application, the Oracle file is found:

262
263 `sakai-src/syllabus/syllabus-app/src/sql/oracle/sakai_syllabus.sql`

264
265 Find the sql files for the applications new to your Sakai, get them modified, and run
266 them.