Installation Guide

Information

This explains the process involved with installing and getting started using the Evaluation System tool.

Installation

The EvalSys group is currently at work on a QA effort for a new release of the EvalSys tool. Those who are interested in evaluating the tool may want to check out trunk until the next QA release candidate is available.

- The current release of evaluation is 1.4.0
    - Requires Sakai 2.6.x or higher
    - Requires Hierarchy 1.2.4 or higher
    - Requires EntityBroker 1.3.5 or higher (1.3.5 is included in Sakai 2.6.x)

- The previous release of evaluation is 1.3.0
    - Requires Sakai 2.5.x or higher
    - Requires Hierarchy 1.2.4
    - Requires EntityBroker 1.3.3 or higher (1.3.3 is included in Sakai 2.5.x)

- I am hoping to simplify this process by building a war file which includes most everything in a single file sometime in the future but the way Sakai is structured makes this quite difficult at the present time.

- **Note:** Currently requires at least Sakai 2.5.x, older versions will not work, current development is against Sakai 2.7.x but older versions probably will work

1. Download the source code from Subversion (choose the most recent tag in most circumstances):
   Tags available here: https://source.sakaiproject.org/contrib/evaluation/tags/
    - If you are upgrading to version 1.2 there are significant DB changes that will require the evaluation tables to be dropped and recreated,
      see the notes below for the commands to drop the tables (we do not have a migration script at this time)
    - If you want trunk then use the trunk directory instead and checkout with subclipse or use this command:
      svn co https://source.sakaiproject.org/contrib/evaluation/trunk evaluation
2. Put the source code into your sakai source directory in a folder called **evaluation**
3. Add jobscheduler project if on a version of Sakai older than 2.4.x (required)
    - **NOTE:** This is only required if you are on a version of Sakai older than 2.4.x, newer
      versions of Sakai include this already, version must be a custom branch called aaronz_2-4-x

    a. Checkout jobscheduler into your sakai source directory (use subclipse or the command below)
       svn co https://source.sakaiproject.org/svn/jobscheduler/branches/aaronz_2-4-x jobscheduler
    b. Build job scheduler using maven (mvn clean install sakai:deploy / maven sakai) from within the **jobscheduler** folder
       See notes below related to the polling job.
2. Add entity broker project (required for Sakai versions older than 2.6, otherwise, you already have it)
    a. Checkout entity broker 1.3.3+ into your sakai source directory (use subclipse or the command below)
       svn co https://source.sakaiproject.org/svn/entitybroker/tags/entitybroker-1.3.6/ entity-broker
    b. Build entity broker using maven (mvn clean install sakai:deploy / maven sakai) from within the **entity-broker** folder
3. Add hierarchy project (required for trunk and eval 1.2+)
    a. Checkout hierarchy into your sakai source directory (use subclipse or the command below)
       svn co https://source.sakaiproject.org/contrib/caret/hierarchy/tags/hierarchy-1.2.4/ hierarchy
    b. Build hierarchy using maven (mvn clean install sakai:deploy / maven sakai) from within the **hierarchy** folder
    c. (maybe not necessary, but works for Sakai 2.6.2 and hierarchy 1.2.4) Adapt the parent version tag in hierarchy/pom.xml to your current Sakai version.
4. If using trunk revision <= 67204 (maybe even for higher revisions) and Sakai 2.6.2. Do this also if you get some strange "dependencies.dependency.version missing" errors.
    a. Run the Kernel K1 conversion scripts from the evaluation directory: http://confluence.sakaiproject.org/display/DOC/Sakai+2.6+Contrib+Tool+Migration+Tips
    b. (maybe not necessary) Adapt the master and parent version tags in evaluation/pom.xml to 2.6.2 (or your current 2.6 version)
        i. evaluation/pom.xml: <sakai.version>2.6.2</sakai.version>
        ii. evaluation/pom.xml: <version>2.6.2</version>
5. As the evaluation system currently depends on Sakai 2.5.4, you may need to update the pom.xml files to reflect the sakai version you are using.
6. Run **maven sakai** from within the **evaluation** folder
    - The evaluation project can be built and deployed with Sakai 2.5.x, 2.6.x, 2.7.x, 2.8.x., or 2.9.x. This is enabled by use of maven profiles. For example, to do a full build of evaluation with Sakai 2.9, you might enter the following maven command on the command line:

      ```
      mvn clean install -D sakai2.9
      ```

    - This will download all needed jars and compile the source code

    - Maven 1 users: You may need to add the following repositories to your maven 1 build.properties to get all needed jars:
        a. http://source.sakaiproject.org/maven/
        b. http://www2.caret.cam.ac.uk/maven
        c. http://repo1.maven.org/maven/
    - For example: Your build.properties should have this line (or something like it)

```
maven.repo.remote = http://source.sakaiproject.org/maven/,http://www2.caret.cam.ac.uk/maven/,
http://repo1.maven.org/maven/
```

7. Start tomcat and check the log for errors, there should be none if all jars were downloaded and installed correctly

> N.B. When installing the Evaluation System in a clustered environment, be sure to start the servers up one at a time to avoid EVALSYS-957 (which would prevent some servers from starting up)!

Upgrading

If you are upgrading from 1.1 to 1.2.0 then you will need to update your table structure. Running with auto.ddl on will update the tables to the correct structure, however, there will be columns which are nullable which you will need to ensure have no nulls. Running this script will take care of them.
eval-1.2.0-migrate.sql

Getting started

1. The tool should be added to all **My Workspaces**for all users
    - You can also add the tool to any Sakai site you like, it can function from any location in Sakai
2. Set the tool permissions for the maintain role (create_template, begin_evaluation, be_evaluated)
    a. Click Realm
    b. Click !site.template
    c. Click Role Id - maintain
    d. Check the following boxes
        - eval.assign.evaluation
        - eval.be.evaluated
        - eval.write.template
    e. Click the Done Button
3. Set the tool permissions for the access role (take_evaluation)
    a. Click Realm
    b. Click !site.template
    c. Click Role Id - access
    d. Check the following box
        - eval.take.evaluation
    e. Click the Done Button

1. #* **Note:** you will need to set the permissions for all existing sites and it is a good idea to set these permissions for the site. template so they will be inherited in any existing sites
2. To grant users admin privileges (to make someone a super admin)
    a. Click Realm
    b. Click /site/!admin
    c. Click Grant Ability at top menu
    d. Enter user id
    e. Select role - admin
    f. Click the Save Button

Notes

- Evaluation currently requires generic-dao 0.9.1. If you have an older version in your tomcat shared then the system will fail to startup. Make sure you only have one copy of generic-dao in shared.
    - If you have more than one copy you will get a stacktrace when tomcat is starting up which indicates a hibernate query exception.
    - If you remove all versions in shared and then rebuild the evaluation tool, the correct version will be download for you.
- You will need to turn on auto.ddl in the sakai.properties file to create the schema
    - Alternatively, you can try using Nuno's script here to create SQL which you can run manually: dbschema-export-task.zip
- The database tables may be cleared by dropping them, in this order
    - Here is the syntax for Oracle:
      drop table EVAL_ADHOC_EVALUATEES;
      drop table EVAL_ADHOC_PARTICIPANTS;
      drop table EVAL_ADHOC_GROUP cascade constraints;
      drop table EVAL_ADHOC_USER;
      drop table EVAL_CONFIG;
      drop table EVAL_LOCK;

drop table EVAL_TAGS;
drop table EVAL_TAGS_META;
drop table EVAL_TRANSLATION;
drop table EVAL_ASSIGN_GROUP;
drop table EVAL_ANSWER;
drop table EVAL_RESPONSE;
drop table EVAL_TEMPLATEITEM;
drop table EVAL_EVALUATION;
drop table EVAL_EMAIL_TEMPLATE;
drop table EVAL_TEMPLATE;
drop table EVAL_ITEMGROUP cascade constraints;
drop table EVAL_ITEM cascade constraints;
drop table EVAL_SCALE cascade constraints;
drop table EVAL_SCALE_OPTIONS;
drop table EVAL_GROUPNODES;
drop table EVAL_GROUPNODES_GROUPS;
drop table EVAL_ASSIGN_HIERARCHY;
drop table EVAL_ASSIGN_USER;

- Here is the mysql syntax:
  set FOREIGN_KEY_CHECKS=0;
  drop table EVAL_ADHOC_EVALUATEES;
  drop table EVAL_ADHOC_PARTICIPANTS;
  drop table EVAL_ADHOC_GROUP;
  drop table EVAL_ADHOC_USER;
  drop table EVAL_CONFIG;
  drop table EVAL_LOCK;
  drop table EVAL_TAGS;
  drop table EVAL_TAGS_META;
  drop table EVAL_TRANSLATION;
  drop table EVAL_ASSIGN_GROUP;
  drop table EVAL_ANSWER;
  drop table EVAL_RESPONSE;
  drop table EVAL_TEMPLATEITEM;
  drop table EVAL_EVALUATION;
  drop table EVAL_EMAIL_TEMPLATE;
  drop table EVAL_TEMPLATE;
  drop table EVAL_ITEMGROUP;
  drop table EVAL_ITEM;
  drop table EVAL_SCALE;
  drop table EVAL_SCALE_OPTIONS;
  drop table EVAL_GROUPNODES;
  drop table EVAL_GROUPNODES_GROUPS;
  drop table EVAL_ASSIGN_HIERARCHY;
  drop table EVAL_ASSIGN_USER;

set FOREIGN_KEY_CHECKS=1;

- **Job Scheduler** - Email notification and evaluation state change based on the passage of time uses a new job scheduler service that depends on polling. Andrew Poland was working on configurable and automatic polling, but last I heard this had not been checked in. (Update: May 9 - Andrew wrote the code to automatically register the job but there is a bug. If it is fixed the code will go in the 015 tag.) Update: Andrew got this done for 2.4.
- With the 2.4 version of jobscheduler there is no need to use the jobscheduler tool to schedule running of the ScheduledInvocationRunner.* The job automatically registers itself on startup. It runs every 10 mins by default.
  - To change that timer you can set:
    - jobscheduler.invocation.interval=600
      Where 600 is the number of seconds between runs.
- Evaluation email and scheduling jobs that are created go into a table SCHEDULER_DELAYED_INVOCATION.
- When the Scheduled Invocation Runner runs, it looks for entries in that table and executed those that have a date equal to or before the current date. When a job is run the Scheduled Invocation Manager will removed its entry in the SCHEDULER_DELAYED_INVOCATION table.
- **Note:** A patch attached to SAK-10611 is needed to prevent multiple instances of Scheduled Invocation Runner running the same command multiple times. This can happen if the polling interval is shorter than the job completion time. EVALSYS-202 and EVALSYS-133 relate to this. Andrew Poland will apply the patch (it changes Scheduled Invocation Runner to a StatefulJob) to trunk and 2.4.x but has not done so as of Jul 19 07.
- **Note:**The Sakai EmailService requires unauthenticated SMTP. If unauthenticated SMTP is not available in the development environment, email can be logged by adding in sakai.properties:

  ```
  # log email rather than send it
  testMode@org.sakaiproject.email.api.EmailService=true
  ```