

Moodle 2.9.1 – Customization: ACCS - Auto-Create Course Shells

Contents

Introduction	2
Installation.....	2
Auto-create Course Shell.....	4
Use cases.....	4
Database changes.....	4
Code changes	4
Auto-update Course Shell	5
Use cases.....	5
Algorithms / Code changes	5
Database changes.....	5
Code changes	5
Testing.....	6
Test Case setup	6
Test Cases.....	7
Appendix A – ACCS process flow diagram	9

Moodle 2.9.1 – Customization: ACCS - Auto-Create Course Shells

Introduction

This middleware solution manages course creation in Moodle (RRU customized Moodle) based on data in the Agresso Student system. The middleware is written in PHP/MySQL and utilizes some of the Moodle libraries to minimize code duplication while maintaining Moodle's business rules. PHP/MySQL solution was chosen as it allows complete customization capability as well as a large base of people who can maintain the code.

Installation

1. MS SQL Server setup – to be performed by DBA
 - a. In the Agresso database (VSS:
\$/AIS_Data/DB_Dev/SQL_2008/Agresso/views/IS_DataMart):
 - a. Execute uvw_DataMart_Learn_NewCourseOfferings.sql
 - b. In the IS_Datamart database (VSS:
\$/AIS_Data/DB_Dev/SQL_2005/IS_DataMart/procedures/mgmt_scripts):
 - a. Execute Moodle-ACCS tables and security.sql
 - c. In the IS_Datamart database (VSS:
\$/AIS_Data/DB_Dev/SQL_2005/IS_DataMart/procedures):
 - a. Execute usp_Update_Agresso_Learn_NewCourseOfferings.sql
 - d. In the IS_Datamart database (VSS:
\$/AIS_Data/DB_Dev/SQL_2008/IS_DataMart/procedures/Moodle):
 - a. Execute Learn_esp_GetNewCourses.sql
 - e. Set up job to synch data between Agresso and IS_DataMart
2. Grab the code from the GitHub repository and clone the repository. Example:
 - a. `cd /var/web/wwwroot/moodle/local`
 - b. `git clone https://github.com/royalroads/accs.git`
3. Rename or copy the accs_config.example.php file to accs_config.php and edit to reflect appropriate settings for the server
 - a. Set connection settings for the MS SQL database server
(Get this information from DBA)
 1. mssqlserver : the name of the MS SQL Server on which the IS_Datamart db is stored
 2. mssqldb : the name of the database where the stored procedures are stored
 3. mssqluser : the name of the database user that will be used to perform the connection to the database
 4. mssqlpwd : the password of the database user that will be used to perform the connection to the database
 - b. Set Miscellaneous ACCS settings
 - a. emailto : ACCS will generate email notifications when courses are created. They go to this address.
 - b. defaultcatid : default category for courses if destination category is not found

Moodle 2.9.1 – Customization: ACCS - Auto-Create Course Shells

- c. log : absolute path to log file for ACCS including the filename
- d. monthsahead : how many months into the future to look for new courses
- e. res_name : name of the default page that is added to each new course shell
- f. res_intro : description of the default page that is added to each new course shell
- g. default_courses : course id's for default courses
- h. admin_user : user id of the administrator account

4. Load course categories to create category structure

The course category structure specified for Moodle 2.0 is pulls Program Type, Department, and Program from the Agresso database to build the structure in Moodle.

a. Set up an ODBC MySQL connector – to be performed by DBA

1. If it doesn't already exist, set up an ODBC MySQL connector to allow the creation of a linked server between MS SQL and MySQL servers.
2. Create linked server between MS SQL and MySQL servers

b. Create/Load the temporary table in Moodle with the data – to be performed by DBA

1. Execute `SQLJob_Deploy_mdlnfinchCategorySource.sql` (VSS: `$/AIS_Data/DB_Dev/SQL_2008/Agresso/mgmt_scripts/ACCS/`)
2. Confirm the data is present by running a query against the table in MySQL client

```
SELECT * FROM mdl_tmp_category_source
```

c. Create the course categories in Moodle

1. Open a browser and navigate to `http://moodle root/local/accs/setup/accsloadcategories.php`
2. The screen should output progress information as it executes
3. Upon completion, copy the page results to a text file and store for installation confirmation purposes – call the file something like 'Moodle 20 Course Category population.log'.

d. Clean up – to be performed by DBA

1. Drop the `mdl_tmp_category_source` table from the Moodle database

5. Install the ACCS plug-in

- a. Log into Moodle using an administrator account
- b. The “upgrade” screen will appear or go to the Site administration -> Notifications to trigger the “upgrade” screen
- c. Click the “Upgrade” button and a screen with “local_accs” Success” should appear
- d. Navigate to Site administration -> Plugins -> Local plugins -> Manage local plugins
 - i. Should see something like - accs | 2012010901

6. Set up cron jobs

Data does not come directly from Agresso, but is instead imported into IS_Datamart via a SQL Server Job every 30 minutes between 5:30 AM and 11:00 PM each day.

a. Auto-create course shell

Moodle 2.9.1 – Customization: ACCS - Auto-Create Course Shells

Add a line to the crontab so that run_accs-createcourses.php is run once a week on Monday at 6:30 am :

```
0 6 * * 1 wget -q -O /dev/null http://hostname/moodle/local/accs/run_accs-  
createcourses.php
```

Auto-create Course Shell

Creates new course shells in Moodle with basic information.

Use cases

1. A new undergraduate course is created in the Agresso Student Information System that has a start date within 9 months of the current date
2. A new graduate course is created in the Agresso Student Information System that has a start date within 9 months of the current date
3. A new Credit-Non-credit course is created in the Agresso Student Information System that has a start date within 9 months of the current date
4. A new course is created in the Agresso Student Information System that has a start date within 9 months of the current date and is part of more than one program within the same department
5. A new course is created in the Agresso Student Information System that has a start date within 9 months of the current date and is part of more than one program across different departments

Algorithms / Code changes

Database changes

- Agresso.dbo.uvw_DataMart_Learn_NewCourseOfferings
- IS_Datamart.dbo.Agresso_Learn_NewCourseOfferings
- IS_Datamart.Learn.usp_GetNewCourses
- Moodle.accs_newcourses table

Code changes

ACCS files are stored in the /local/accs folder

- /db/install.xml – db tables to be installed
- /db/install.php – any install related tasks
- accs_config.php – stores configuration information specific to ACCS middleware
- accslib.php – accsware specific functions
- database.php – database specific functions
- mssql.php – Microsoft SQL Server specific functions
- run_accs-createcourses.php – simple php script to allow the CRON to run ACCS
- dev_notes.txt – stores content for the Development Notes resource.
- version.php – Moodle version management file

Moodle 2.9.1 – Customization: ACCS - Auto-Create Course Shells

Auto-update Course Shell **(THIS FEATURE HAS BEEN DISABLED)**

Updates existing Moodle courses that have not started based on changes from Agresso.

Use cases

1. The start date of an undergraduate course is updated to a future, but not more than 9 months into the future
2. The description of a course offering is changed for a course that has a start date no more than 9 months into the future

Algorithms / Code changes

Database changes

- Agresso.dbo.uvw_DataMart_Learn_NewCourseOfferings
- IS_Datamart.dbo.Agresso_Learn_NewCourseOfferings
- IS_Datamart.Learn.usp_GetNewCourses
- Moodle.accs_newcourses table

Code changes

ACCS files are stored in the /local/accs folder

- /db/install.xml – db tables to be installed
- /db/install.php – any install related tasks
- accs_config.php – stores ACCS middleware settings
- accslib.php – accsware specific functions
- database.php – database specific functions
- mssql.php – Microsoft SQL Server specific functions
- run_accs-updatecourses.php – simple php script to allow the CRON to run ACCS
- version.php – Moodle version management file

Moodle 2.9.1 – Customization: ACCS - Auto-Create Course Shells

Testing

Note: Requires access to Agresso client. Knowledge of Agresso is an asset to understanding the following tests.

Test Case setup

In the Agresso client (alpha or beta)

1. Create a new course offering that has a start date within the next 9 months
 - a. Add the course to a program offering structure that will start in the near future
 - b. Make note of the following:
 - i. Course title
 - ii. Course Start date
 - iii. Course Level
 - iv. Program code/Program Offering Start date
 - v. Ed depart code (on the Relation tab in the Reporting folder of the Program template)
 - vi. Delivery mode (Program offering)
2. Update the start date and description of an existing course offering that has already been imported into Moodle
 - a. Open a course offering and update the offering description (offering_desc) and the start date so it one month into the future
 - b. Make note of the following:
 - i. Course title
 - ii. Course Start date
 - iii. Course Level
 - iv. Program code/Program Offering Start date
 - v. Ed depart code (on the Relation tab in the Reporting folder of the Program template)
 - vi. Delivery mode (Program offering)
3. Create a new “shared” course offering that is shared between 2 programs that have the same department.
 - a. This is accomplished by including the course in the Program Offering structure
 - b. Make note of the programs and department
4. Create a new “shared” course offering that is shared between 2 programs that have different departments.
 - a. This is accomplished by including the course in 2 different Program Offering structures
 - b. Make note of the programs and departments

Moodle 2.9.1 – Customization: ACCS - Auto-Create Course Shells

Test Cases

1. New course
 - a. Run the ***run_accs-createcourses.php*** to create new courses that have been created in Agresso
In Moodle, the new course that was created in Agresso (Step 1 of Test Case setup) should:
 - exist under the correct program category
 - has a “Development Notes” page resource included that opens in the current page
 - confirm course settings
 - have the correct course full name format
CourseCode [ProgramCode ProgramStartYear-Section DeliveryMode] course start date – course description
 - have the correct course short name format
CourseCode [ProgramCode ProgramStartYear-Section DeliveryMode] course start date
 - have Course ID number of “*course code__course offering code*”
 - have Format = “Topics format”
 - have Course start date the same as Agresso record
 - have Availability = “This course is not available to students”
2. New shared course (within same department)
 - a. Handled by 1.a Run the “run_accs-createcourses.php” script
The new course should have same confirmations as Test 1, but the Names should be different:
confirm course settings
 - have the correct course full name format
CourseCode [DepartmentCode ProgramStartYear-Section DeliveryMode] course start date – course description
 - have the correct course short name format
CourseCode [DepartmentCode ProgramStartYear-Section DeliveryMode] course start date
3. New shared course (across departments)
 - a. Handled by 1.a Run the “run_accs-createcourses.php” script
The new course should have same confirmations as Test 1, but the Names should be different:
confirm course settings
 - have the correct course full name format
CourseCode [DepartmentCode ProgramStartYear-Section DeliveryMode] course start date – course description
 - have the correct course short name format

Moodle 2.9.1 – Customization: ACCS - Auto-Create Course Shells

*CourseCode [DepartmentCode ProgramStartYear-Section DeliveryMode] course
start date*

4. Update existing course start date/description (we do not run this script anymore)

- a. Run the ***run_accs-updatecourses.php*** to update the course in Moodle with appropriate data

Confirm course settings:

- have the correct course full name format

*CourseCode [DepartmentCode ProgramStartYear-Section DeliveryMode] course
new start date – new course description*

- have the correct course short name format

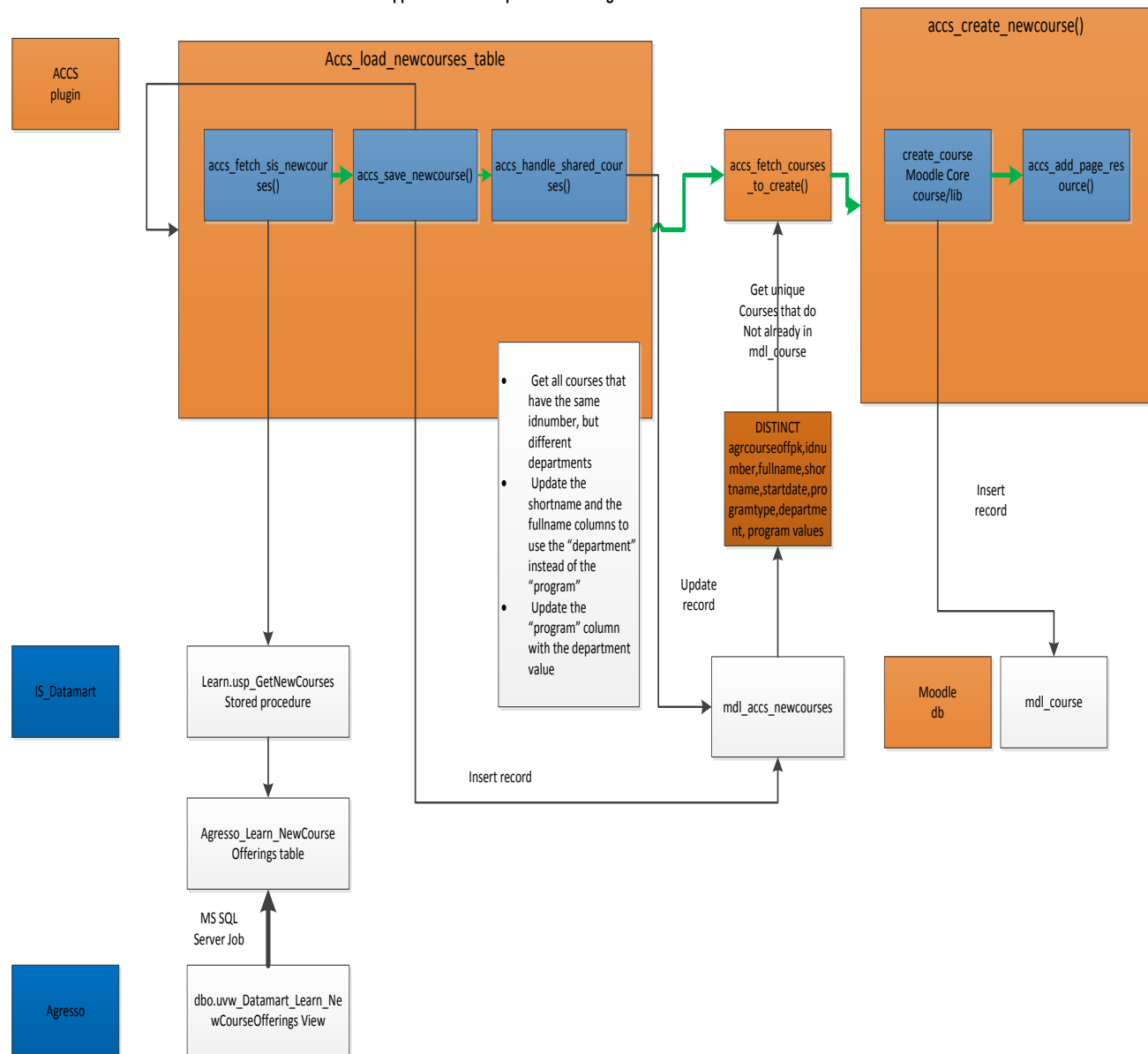
*CourseCode [DepartmentCode ProgramStartYear-Section DeliveryMode] course
new start date*

Course start date the same as Agresso course offering

Moodle 2.9.1 – Customization: ACCS - Auto-Create Course Shells

Appendix A – ACCS process flow diagram

Appendix A – ACCS process flow diagram



Note: Moodle Course Shell Creation

Located in: O:\Staff\academic and information services\it-services\public\Projects\2015-2016\ Moodle 29 Upgrade\Documentation\Plugins