

Overview

A Quick Technical Guide To LAMS

This is an overall guide to how LAMS works for new developers. It does not explain all the details of how LAMS works, but should give you enough background so that you can understand other information on the wiki.

There is also a [teachers guide](#) (PDF) available.

See Also [Glossary of Terms](#).

LAMS allows the user to create and participate in learning designs, also known as sequences. A learning design is a sequence of activities. A LAMS learning design has two components - the structure of the learning design (ie the order of the activities) and the content (the heading, description, questions, links, etc).

Users in LAMS are assigned to permanent Groups. Groups can have sub-groups. These groups related to "real world" groupings such as a course, tutorial class, etc.

A lesson is a learning design that run with a class. When a lesson is started, the user selects a learning design then selects users from a Group. The selected users form a new lesson class.

While the lesson is running, there can also be temporary groups created that apply only within the lesson. These break the lesson class into smaller groups as directed by grouping activities within the learning design.

LAMS is broken into two parts - the core and tools. The LAMS core manages the structure parts of learning design structure: the order of the activities, allocating users to groups, managing when users can progress to particular activities, etc. The core also takes care of logins, system administration, etc.

Tools are self-contained modules, which form most of the "functionality" that the Learner interacts with in LAMS. For example, the Question and Answer screens come from the Question and Answer tool. The Survey screens are part of the Survey Tool.

Typical LAMS interactions

Author creates a learning design

User View:

An author (usually a teacher) logs into LAMS and brings up the authoring interface. This shows all the tool available. The teacher can import a learning design from the LAMS community, open an existing learning design or start a new learning design.

Working in the authoring interface, the author sets up their design. They click on a tool and drag it into the layout window, creating an activity. The author double clicks on the activity icon to open up the tool's authoring screen. The author can set the activity to be "define later" i.e. the content will be defined later or "run offline".

When the author is finished, the author saves the design.

Technical View:

The main part of authoring interface is a Flash movie which is part of the LAMS core. When the user double clicks on an activity, the Flash movie gets a new tool content ID from the server and then calls a URL supplied by that tool to open the tool's authoring screen. When the tool's screen is closed, the tool stores the content for the activity.

When the user saves the entire design, the Flash movie sends the structure back to the server and the design is saved by the core modules. Note: the tools create their content data before the core saves the overall sequence.

Teacher creates a lesson

A teacher log into LAMS and brings up the monitoring interface. This shows all the current lessons and allows the teacher to start new lessons.

To start a lesson, the teacher selects an existing learning design and a group. The list of learners and staff members in that group is displayed and the teacher selects the learners and staff that is wanted for this lesson. The selected learners and staff go into a new lesson class.

The LAMS core also duplicates the learning design. During the copy, the content of each activity is copied - the core calls each tool and asks the tool to copy the content.

Then the lesson is created and linked to the new lesson class and the copy of the learning design. As it is linked to a COPY of the learning design, the teacher can go into authoring and change their original learning design if they want.

Teacher manages a lesson

Once a lesson is started, the teacher can manage the lesson. Again, the main part of the monitoring interface is a Flash movie. The Flash movie shows the structure of the lesson. When the teacher selects a particular activity, the Flash movie calls a URL supplied by that tool to open the tool's monitoring screen. Depending on whether the monitoring screen should related to the content (for changing content) or should related to what the learners are doing, the URL will contain with the tool content id or the tool session id(s).

Learner participates in a lesson

Once a lesson has started, the learners can participate in a lesson.

When the learner logs into LAMS, the learner interface is started. Again, the main part of the interface is a Flash movie. The learner selects the lesson that they wish to participate in. The Flash movie calls the progress engine (part of the LAMS core) to calculate which activity the learner should do.

The Flash movie calls a URL based on URL supplied by the tool. The learner then interacts with the tool. This may involved one or more screens. When the user has finished the activity, the tool notifies the progress engine that the user is completed and the progress engine calculates the next activity.

In the process of checking what a learner should do, the progress engine also creates a tool session for each tool activity/group. A tool session represents the use of a tool for a particular activity for a group of learners. So if an activity is "ungrouped" (ie no grouping applied in authoring) then one tool session will be created for the tool activity. If the activity is grouped, then there is one session for each group. So one tool content id can related to one or more tool sessions.

When a learner reaches a tool activity for which no applicable tool session the progress engine will generate a tool session id, and will call the tool to create the tool session id.

Export and Import

Authors can export and import learning designs in XML format When an export occurs, the LAMS core exports the structure of the design, and calls each tool to create XML describing the content of the activities. The import is similar.

The export is used to support exporting the Learning Design in IMSLD format.

Teachers and learners can export what has been seen by the students using the portfolio export. This forms a record of the lesson. Again, the core takes care of the structure and calls the tools to generate the text for each activity.

LAMS Deployment

LAMS 2.0 runs on JBOSS application server. See [System Requirements](#) for full system requirements and supported browsers.

LAMS is deployed as a J2EE 1.3 compliant EAR in exploded format.

This EAR contains jars and web modules for the core lams functionalities (e.g. Author, Monitor, Learner, Admin etc.) and third party library jars that are required by the core components. The third party jars are included in the EAR classpath, so the core modules and tools can assume that they will be available. See [Common Java Libraries](#) for more details.