ceLTIc

Creating Environments for Learning using Tightly Integrated Components
A formerly JISC-funded project

celtic-project.org

RATING INTEGRATIONS FROM LOVELY TO INCREDIBLE

R | ROBUST ■ CERTIFIED ■ *TRUSTS TOOL CONSUMER* ■ PROPERLY SIGNED ■ *STRONG SECRET* ■ MINIMUM REQUIRED PARAMETERS ■ *PRIVACY* ■ SSL | ADAPTABLE ■ FLEXIBLE ■ *CUSTOM PARAMETERS* | TRANSPARENT ■ SURVIVES COURSE LIFE-CYCLE: CREATE – COPY – MOVE – ARCHIVE – RESTORE | EASE-OF-USE ■ *SEAMLESS* ■ SIMPLE TO INSTALL: XML – DOCUMENTATION ■ *ROLE ALIGNMENT* ■ INTUITIVE ■ *NO SCROLLBARS*

A
T
E

lti.tools/rate

LTI: Rating Integrations from *L*ovely *To* *I*ncredible

As successful and useful as LTI® (Learning Tools Interoperability®) has become as a mechanism for extending on-line course environments, there is still scope for differences in the integrations which have been implemented which can impact users in different ways. This document describes the areas where such differences may occur and highlights how LTI integrations may be evaluated to identify those demonstrating a higher level of quality.

For convenience the areas (or categories) have been divided into the following four groups (having the acronym of RATE):

- **R**obust
- **A**daptable
- **T**ransparent
- **E**ase-of-use

The intention is to be as comprehensive as possible, but no doubt some important factors have been omitted for later addition! If nothing else, they should provide a source of useful questions to ask tool providers about the way in which their integrations work. How does yours measure up?

Robust

Probably the most important aspect of an LTI integration is that it works reliably; this category explores what makes this the case.

Certified

One of the most definitive marks of quality is that the integration has passed the IMS certification tests. These tests are available for IMS members and drive products through an exhaustive process to seek any inherent weaknesses in their implementations. Of course, as semi-automated tests their main focus is on the mechanics of the specification, but a certified product should indicate that it is built on a robust architecture.

Trusts Tool Consumer

Underlying an LTI integration is a trust relationship between the two parties (a tool consumer and a tool provider) as evidenced by a consumer key and shared secret. The normal expectation is that a tool provider should trust the data it receives from the tool consumer in launch messages. This includes details of the user (e.g. their name) and the role they have in the course from which a launch originates. A good tool provider will trust such data and anticipate that it could change at any time. The main conflict for tool providers arises when they have access to similar data from another source – they must then decide which one to rely on when there is disagreement (e.g. a role is different or a launch request comes from a previously unknown user). Since the launch data is accurate at the time of the user activity, this should typically take precedence.

Properly Signed

The data in messages and service requests is signed using the shared secret. This is a well-defined process using OAuth 1.0A for which many standard libraries are available. However, there can be stumbling points in generating signatures on the part of both the tool consumer and the tool provider (for verifying the signatures it receives). A quality implementation should have a robust signing process which works under all conditions so that messages are not rejected causing a service failure.

Strong Secret

The signing of messages is an essential part of securing the data being passed between the two parties, but it can only be relied upon by the recipient if the secret used to generate the signature is secure. This is a combination of having processes in place to ensure that only the two parties know the secret being used and also using a secret which is both long enough and complex enough to make it hard to break. In addition a procedure for changing the secret from time to time can add to the strength of the security it provides.

Minimum Required Parameters

Very few of the parameters passed in an LTI launch message are required – most are either recommended or optional. The certification process also ensures that tool consumers are capable of enabling and disabling the passing of personal data. Thus, a tool provider should make as few assumptions as possible about the data it receives on launch and, unless a parameter is vital to providing a useful service, should seek to cope with the absence of data it would have liked to know. For example, rather than denying a service because a user's name is not passed, a tool provider could invent a name (such as "User <user_id>") and leave it for the tool consumer to turn on the passing of names if it finds this inadequate.

Privacy

The privacy of data is very important in education, especially in relation to personal data. A tool provider which requires personal data to be passed in a launch message should be able to comply with any privacy policies of the tool consumer. Furthermore, it should not require data to be passed if it is not essential to the service it is offering.

SSL

Although the launch data passes through a user's browser, it is always recommended that tool providers use SSL so that the data is encrypted "over the wire". This is especially important if the message includes personal data about the user. It is no different from the recommended practice for on-line course environments to use SSL-based servers, and a similar requirement for tool providers will also avoid any possibility of mixed content when they are opened within the same browser window (which many browsers deny by default).

Adaptable

An on-line course environment is expected to able to adapt to the individual needs of institutions, and this should be extended into functionality offered by LTI tool providers even though they are hosted externally. The extent of the requirements will depend largely on the nature of the tool provider, but developers should always consider the environment with which they are being integrated and the types of differences that may exist (both within a single tool consumer and across multiple tool consumers).

Flexible

Flexibility must be interpreted in relation to the tool provider concerned, but here are a few considerations which may be relevant:

- does the tool provider map user roles in a similar way as they are used in the tool provider (especially, for example, a teaching assistant role which may have quite different privileges in different institutions)?
- is it possible to launch into the tool provider from the locations within the UI of the tool consumer which are deemed most appropriate by users?
- is the tool provider able to accommodate different choices regarding the launch data it is passed?

Custom Parameters

The LTI specification allows additional parameters to be passed in a launch; these can be defined by the tool provider and provide a simple mechanism by which it can allow customers to tailor their experience. This could be used for something as simple as a URL to use for help links within the tool provider's UI, or the word to use to describe a particular type of object (such as a course or module). All tool consumers should allow custom parameters to be specified so, whilst not being the slickest mechanism for specifying choices, it can be used to avoid the need to create a configuration page on the tool provider system for instructors and/or system administrators to use. For example, the LTI connector for WebPA 2 supports a number of options for overriding the interface, including specifying an alternative logo (see http://celtic.lti.tools/wiki/LTI/Best_Practice/Issues_for_Developers#Override_interface).

Transparent

Transparency is closely linked to adaptability – the main aim here is to ensure that users are hardly (if at all) aware that the tool they are accessing is external to their learning environment. Some of the techniques described above can allow the integration to appear seamless in use, but there are other workflows which are important to instructors and course administrators which are also important to consider. Adding links to courses is a basic requirement, but tool providers should also consider how well these links survive when courses are copied/archived/restored etc.

Survives Course Life-cycle: Create – Copy – Move – Archive – Restore

It is quite common for an on-line course to be repeated within the same academic session and/or in future academic sessions. Most instructors appreciate the ability for re-using their courses with a minimum of effort, hence learning environments commonly provide mechanisms for copying or resetting courses, perhaps even updating all the dates for the activities they contain. Similarly, it is important to be able to archive/backup and restore courses in the event of some unfortunate event. All these actions should lead to questions about what happens to LTI links in a course when they occur.

Of course, what happens is dependent upon the nature of the service being offered, but that does not remove the need for the question to be addressed. When an LTI link is copied it will be assigned a new ID (resource_link_id parameter value). Thus, any association between the old resource ID and a workspace, activity, content item, etc. in the tool provider will be lost if this is based on the ID value being passed. But in some cases, an instructor may expect the association to be maintained or, at least, a copy of the association to be applied to the new link. For example, consider the following tools:

- *WordPress* – the LTI integration for WordPress available from http://www.spvsoftwareproducts.com/php/wordpress-lti/ will create a new blog for each new resource link which is launched from. This behaviour is probably as expected, it is unlikely that the content of the old blog would be needed, instead the new link (perhaps a copy in a new version of the course) would start afresh.
- *Scribblar* – is similar to WordPress in that new links are merely associated with a new virtual classroom. However, it may be helpful for any configuration settings to be replicated from the old link so that the room options remain the same (though the current implementation does not do this).
- *Atomic Learning* – Atomic Learning provides instructors with the ability to set up a playlist of training videos for students to access. One link, for example, could provide training in Microsoft Excel. If the course is copied an instructor would no doubt expect the new copy of the link to continue to be associated with a playlist of videos on Microsoft Excel, rather than ask them to set-up a new playlist. A best practice feature exists which provides a standard way for a tool consumer to pass through the resource ID from which a new link has been copied, thereby allowing a tool provider to duplicate an existing configuration when it encounters a new ID for the first time. In the case of Atomic Learning this allows the instructor to have their playlists maintained across the course copy process, but for them to then be independent of each other.

Ease-Of-Use

Making a tool which effectively is being plugged into an on-line course in a standard way, easy to use can be a challenge. This category considers some of the criteria which might be used to identify a tool which is easy to use.

Seamless

Ideally a user should not be aware that the tool being accessed is any different from those tools which are native to the learning environment (tool consumer). LTI allows the user to be redirected seamlessly from a link in the tool consumer to a tool provider page by securely passing sufficient information about the user, the context they are coming from and their role in that context to enable the tool provider to establish an appropriate environment for them “on-the-fly”. However, any tool provider for which this data is insufficient is likely to interrupt this flow; for example, by asking users to authenticate themselves separately with the tool provider. This may only occur once per browser session, but it can impact usability. A tool which is able to receive all the data it requires from the LTI launch message is likely to provide a more seamless end-user experience.

Simple To Install: XML – Documentation

For the tool consumer system administrators ease-of-use will relate to the addition of tools within their system. Tool providers should provide clear documentation on what data they require to be included in launch messages, and what data they would use if it is included but which is not required. This allows system administrators to make informed decisions about the configuration of tools. In addition, the documentation may also provide guidance on where it makes most sense to make the tool available within the tool consumer UI; for example, does it support launches from outside course contexts, what other message types are supported (e.g. Content-Item). The LTI specification includes a format for defining tools in XML; if a tool consumer supports this, making an XML definition available can greatly simplify the installation task for system administrators.

Role Alignment

On-line course environments typically support multiple roles for users; the main ones are instructor and learner (or equivalent terms such as teacher and student), but may also permit users to have roles such as content developer, teaching assistant, course administrator, observer and guest. It is not necessary for a tool provider to implement a diverse set of roles, but it is important that when users launch into a tool they are presented with access to functionality commensurate with the role they have been given by the tool consumer. This may be just a matter of clearly describing which LTI roles a tool supports so that tool consumers can appropriately map their roles onto this set, but not all tool consumers offer this facility, so some means for customers to specify this mapping within the tool provider interface may be needed if a tool's roles extend beyond simple instructor and learner cases.

Learning environments also commonly allow an instructor to preview their courses as if they were a student. When doing so, one would expect any LTI launches made to be undertaken with a role of Learner. But this type of Learner is rather different from a normal student and the distinction may be important for a tool to know; for example, so that their results are not included in class averages. Furthermore, in Blackboard Learn 9, for example, it is possible for a new student account to be created every time an instructor enters the preview mode which could lead to a tool being overwhelmed with IDs for users who are not "real". If both parties support the best practice guidance for users with transient roles, then this situation can be avoided.

Intuitive

Tools are easier to use if users find them intuitive. In this case, it is useful for a tool provider to follow conventions adopted by the tool consumer to make it, for example, look and feel as much like the host system as possible. This can include the layout of screens, use of similar widgets for data entry (such as for dates), using the same language/terminology. Achieving this level of integration demands additional effort on the part of tool providers, but this may be justified for tool consumers which are in common use by customers. The ability to identify which tool consumer is being used can be identified from standard launch parameters, and supporting the CSS parameter can assist with providing a similar appearance.

When an LTI link is first added to a course, the tool provider is normally unaware that this has occurred. It is not until an instructor has launched the link that the tool provider sees the resource ID for the first time and can provide an interface for selecting or configuring the connection as appropriate. This is quite different process from the one used to add an internal tool to a course. However, the new LTI Content-Item message overcomes this by allowing a tool provider to be engaged in the process of creating links within the tool consumer. This means that when the link is added to the course it has already been configured and is ready for use; a much more intuitive workflow.

No Scrollbars

Since LTI tools are delivered from systems external to the tool consumer this will inevitably involve the use of frames or iframes if their content is to be displayed within the same browser window. Without careful management, this can lead to duplicate scrollbars appearing on the screen when the content of both the outer and inner windows (frames) extend beyond the dimensions of the display. This is because the tool consumer is not aware of the size of the content within the tool provider controlled frame. Eliminating such unsightly and unhelpful duplicate scrollbars requires a level of co-operation between the two systems such as offered by Canvas by Instructure which allows the tool provider to notify the tool consumer what size iframe it requires.