



# Packaging multiple tools into an LMS

One of the things I'm seeing when I read about Enterprise learning systems is the need to integrate different tools we use under the LMS umbrella to provide a one stop shop for how providing learning content.

Note that where I provide names of products they are the ones I'm most familiar with, they don't imply endorsement or preference, just familiarity.

## User authentication

At the very least we need an [OAuth](#) provider to make sure that users authenticate and that we provide targeted services.

## Messaging and asynchronous communication

The most basic communication tools are threaded discussions and internal email. People still use tools like [Reddit](#) and [Hacker News](#)

In addition, or maybe instead of, internal messaging systems we need to provides way to contact users outside the system using SMS messaging (using [Twilio](#) or a similar tool) and email (with [Sendgrid](#) or equivalent).

The idea is that we can use these tools to notify participants when an activity is due or if they've failed to complete any given tasks.

## Video Conferencing

This is the hardest tool, in my opinion, to implement well. It is possible to create multipoint video conferencing solutions like Zoom but it's not trivial to do so with open source tools. [WebRTC](#) is, in my opinion, the best tool for multipoint, multidevice video conferencing. It also handles text-chat and documents using a [data channel](#).

This is also what makes WebRTC hard to work with from my perspective. Because it does so much that it needs a lot of additional technology to bridge all users and all connection types.

## Surveys and Evaluations

Evaluation is also tricky as it can encompass a large number of ways to evaluate performance and knowledge. I'm not a fan of exams but realize that they have their place. We need some sort of tool create secure exams to measure knowledge; we also need a way for learners to submit projects or to link to projects hosted elsewhere (like [Github](#) or [Codepen](#)). This raises academic honesty issues but I think we're better off discussing them elsewhere.

We need a tool like [Survey Monkey](#) or an in-house solution for survey administration that allows either embedding the survey into the learning system or provides an API to load them locally or create them programmatically.

## Creating content

In professional trade publications, one of the biggest issues that people speak about when asked how can the LMS be improved was the need to create authoring tools inside the LMS.

Creating HTML-based content is easy. There are enough editor applications available that we could take this editor in multiple directions:

- A code editor to write HTML, CSS and Javascript
- A Markdown Editor that can generate HTML
- A WYSIWYG editor that will create HTML under the hood.

Creating the content is part of the equation. Whether it's SCORM or CMI5, you need to package the content and provide the basic package file so the LMS and the corresponding player know how to treat the content.

The other option is to create the content offline and then automate the upload to the LMS.

# Learning APIs

- [SCORM](#) for course packaging
- [xAPI](#) and [cmi5](#) to expand the scope of the learner activities we track
- Integrating an [LRS](#) to expand the learning experiences we track