



30 Sep 2006

30
Sep
2006

» Oracle, Sakai, and the LMOS--Part 1

It's been a while since I've written a post about the notion of a Learning Management Operating System (LMOS). There's some news out of the Sakai world that's as good an excuse as any for doing returning to it.

Apparently Oracle, with the help of Unicon, is going to do quite a bit of work on Sakai in the form of something they call their "Academic Enterprise Initiative" (AEI). According to Oracle's white paper on the topic, the work includes (but is not limited to) the following elements:

- Building batch and real-time integration between Sakai and Oracle's student information system (SIS)
- Creating flexible integration between Sakai and Oracle's teaching-related products (most notably their portfolio and gradebook).
- Making it possible to swap in Oracle collaboration applications (e.g., their discussion forum, calendar, chat, etc.) to replace the analogous Sakai tools
- Presenting the whole thing through Oracle portal or any other JSR-168-compliant portal (like uPortal, for example)
- Creating a student data hub that allows for disparate applications to share data while providing a single "source of truth" to avoid duplicate and conflicting content versions

There's a lot here to unpack. In this post I'm going to look at some of the specifics of what Oracle has in mind and why this advances the LMOS cause significantly--even for those institutions that will never use a single Oracle product. In a follow-up post, I'll drill down into some of the specifics of Oracle's approach and their implications for teaching and learning.

The rationale for AEI that Oracle puts forth in their white paper starts with the observation that the academic software market is littered with lots of special-purpose, stand-alone software systems such as LMS's, test engines, grade books, ePortfolios, SIS's, research management systems, library systems, and so on. And despite the fact that many of these systems serve overlapping needs for overlapping constituencies, they tend to stand alone and work together in support of the needs of students, faculty, and administrators only poorly and only with a great deal of pain:

Because each of these solutions has generally been approached to fill the needs of a particular department or segment in an institution, each solution tends to have been designed and delivered individually with only secondary thought given to the integration or interaction of one system with another. This trend has been exacerbated by the fact that few vendors offer products in more than one space or segment of an institution. A natural result of this specialization is that these vendors have few resources and/or limited expertise available to integrate with

the products from another vendor. This type of integration is especially difficult if, as was often the case in the past, these solutions have taken the form of point-to-point or dedicated, hard-coded solutions that integrate with and only with a particular vendor. Developing and maintaining multiple point-to-point integrations quickly becomes a complex and resource-intensive exercise for these niche solution providers and the higher education institutions they serve.

These factors have resulted in a patchwork of systems, many with minimal data integration, where data is transferred in a batch processes, not through real-time data synchronization, and leaving very little ability to analyze or understand the relationship of data across systems.

There are lots of places where students and faculty experience the pain of this problem, e.g.,

- Getting a course in your LMS set up with all and only the students in the class having proper access is often somewhat less than timely and trouble-free
- A lot of good content never makes it into the students' portfolios because it's too much effort to put it in
- The school provides a great test engine but you have to manually enter the grades into your gradebook
- The grades in your LMS gradebook have to be manually re-entered into the SIS in order to officially report students' grades
- The calendar you use for your class inside your LMS doesn't talk to the calendar you use for everything else, leaving you with two calendars to check

While most of these hassles are not directly pedagogical impediments, they nevertheless tend to reduce the quality of the educational experience by forcing students and teachers to spend time wrestling with their software instead of teaching and/or learning. Oracle proposes to solve this problem by using a service-oriented architecture (SOA) to orchestrate communications among the pieces and a portal for a unified presentation to the users. Regular readers will recognize this as exactly the architecture that my colleagues and I have been promoting as the basis for the LMOS. Oracle also proposes using a data hub, which is a new piece to the LMOS puzzle that I'll get into in more detail in my next post.

At any rate, one of the core ideas behind the LMOS is that the integration challenge Oracle describes is also something we should think about at a more granular level within the teaching environment. You want to be able to have many, many specialized applications for teaching math, physics, creative writing, game theory, scientific method, cellular genetics...get as granular as you need to for teaching purposes. If you need to teach in a specialized way, then you should be able to support the niche teaching method with a niche teaching tool. Which means that our teaching environment becomes one giant interoperability problem. And if you add in a student's perspective as well, e.g., letting them have more control over their own content and not locking it into the course instance, then the problem gets *really* hairy. So what you want from your system is a clean, easy interface for plugging new tools in--but that also doesn't create a chaotic jumble of siloed tools in the process.

That goal turns out to dovetail nicely with Oracle's commercial needs. Among other things, they want to sell their collaboration suite. And they think that the collaboration tools they provide may just be better (in some cases, at least) than the tools provided

by the LMS. At the very least, you get some benefit by using the same tools inside and outside the learning environment. Who wants to have to maintain two separate calendar? (I'm told that Oracle's academic calendaring system is one of the most widely used in higher education.) Or learn two different discussion forum interfaces? And it may very well be that their tools *are* stronger in some cases for some purposes. (I don't know; I've never used their collaboration suite.) At any rate, regardless of how good or bad their tools are, they are motivated to foster the growth of an LMS where it is as easy as possible to swap in non-bundled tools. This is one of the reasons they chose to use Sakai as their LMS:

Sakai is intended to be more than merely a course management system; it is planned as a Collaboration and Learning Environment (CLE). The Collaboration aspect would be embodied in several Sakai tools, including Resources, Chat Room, Email Archive and others. These planned tools are obviously specific to Sakai and are not designed to replace enterprise-level implementations of content management applications, instant messengers, discussion forums, or others.

As noted above, one of the strengths of Sakai would be its ability to accommodate the inclusion of a wide set of tools, assuming a proper set of interfaces or providers exists or is created within Sakai. This is planned to include the ability to switch off a native Sakai tool and replace it with a corresponding tool, whether that tool comes from a commercial vendor, is developed internally, or created as part of another community project.

Oracle Collaboration Suite, which is planned to meet the communication and collaboration needs of enterprise implementations, could be plugged into Sakai when an institution wants to broaden the scope of a deployment to either take advantage of enterprise-level features and scalability by opening it up to a larger segment of the institution's population, or simply to pull together multiple tools and implementations under a single application. For example, an institution could roll-out a single, unified email system across campus so students and faculty could have access both to their institution-wide email system and the Sakai discussion forum messages through their Collaboration Suite email client. [Note: The Oracle Collaboration Suite's forum apparently allows you to view discussions via IMAP, making the integration between discussion fora and the email interface particularly seamless. Sounds kinda nifty to me.]

So AEI really does seem to advance an LMOS-like architecture on a lot of levels. Now, that would be intellectually interesting but not particularly exciting if all of the interoperability interfaces were proprietary. But what's really exciting to me about this initiative is Oracle's heavy emphasis on using current standards whenever possible and working with the community to augment standards or create new ones whenever necessary. Here's just one example of several from the white paper (this one from a section entitled "Commitment to Standards"):

Language only works when all the people trying to communicate speak and understand the same one. The Oracle AEI, by its nature, is an exercise in integration. In order to build upon existing knowledge, participate in the community of users and developers, and ensure future maintainability and extensibility; the AEI will make use of existing standards where it can, and participate in discussions to extend, modify,

or create standards and specifications where needed. This approach includes using IMS Enterprise specifications and actively participating in IMS and the IMS Technical Advisory Board and organizations such as the Sakai Course Management Working Group that has been convened to design a Course Management API to delineate the integration between Sakai and student information systems.

References to standards and working with the community are all over this document. If Oracle lives up to their word, (and even their detractors tend to give them credit for being very good about standards adherence), then the resulting system should play well with many, many different components, whether or not they happen to have been built by Oracle. Oracle's commercial gamble is that, because they have a complete stack of core applications, universities will tend to prefer a single-source vendor. But there would be nothing in the technology to force this. It would truly be a learning management operating system.

Time will tell. In the meantime, they certainly have given me more to think about by adding the element of the student data hub. More on that in my next post.

Posted 2 weeks, 5 hours, 36 minutes ago in Tools, Toys, and Technology (Oh my!) | LMOS and has 0 comments