Universitat Oberta de Catalunya

Next Generation Learning Architecture

Jeff Merriman - birdland@mit.edu
Office of Digital Learning, Strategic Education Initiatives
Massachusetts Institute of Technology

Tom Coppeto - tom@coppeto.org
EagleApps Student System Project
Boston College

Francesc Santanach - fsantanach@uoc.edu
eLearn Center
Universitat Oberta de Catalunya

Cole Shaw - cjshaw@mit.edu
Office of Digital Learning, Strategic Education Initiatives
Massachusetts Institute of Technology

Xavier Aracil - xaracil@uoc.edu
eLearn Center
Universitat Oberta de Catalunya

Index

Executive Summary

Over the past decade and a half, the Learning Management System (LMS) has emerged as the enterprise solution for the delivery of education. An LMS is a single isolated system for supporting the rudimentary functions of online education. However, educational institutions are expanding beyond what a single system can deliver. There is a growing need for diverse educational functions such as advanced assessment, accreditation, personalization and learning pathways, and analytics. Newer software architectures need to offer a wider range of functions and support newer learning methodologies and educational business models. This report examines a software architecture to create and support these next generation learning environments.

Teaching and learning methodologies have and will continue to evolve – yet attempting to predict the future is not without risk. While we can describe the *state of the art* today, we cannot afford to design tomorrow's systems based on today's way of doing things. A next generation learning architecture needs to be able to adapt to new methodologies and educational models as they are developed from ongoing research and practice. The risk of developing upon a rigid architecture is an investment in obsolescence.

Educational technology has been treated as stand-alone product, connected to the rest of an institution's enterprise system via limited data feeds. As a consequence, learning systems have poorly replicated information about students and courses and done so with limited features and capabilities. Next generation learning environments take a different approach that enables richer educational experiences – they treat the business of learning as an enterprise asset on par with student systems, financial systems, and personnel systems. They achieve this using an integration strategy that supports real-time access and dissemination of information and analytics in both directions across the entire enterprise.

The ultimate goal of next generation digital learning environments is to foster a growing and competitive marketplace of educational applications. This marketplace will support consumer choice of educational tools and systems as well as content. Openly published service specifications define the integration strategy with the educational enterprise. Integration is an essential principle to allow these applications to work together effectively to meet the evolving needs of teachers, students and institutional leadership while offering more meaningful support for diverse and advanced teaching and learning methodologies.

This document provides an overview of an architecture which prioritizes integration among all software components to allow the implementation and configuration of each component to change. There are many educational issues that do not depend on technology. Nonetheless, this approach insures that next generation learning environments will be stable yet flexible enough to accommodate changing functional thinking and evolving learning methodologies and business models.

Architectural Overview

One clear truth is that the use of technology in education today is different now than it was ten years ago, and it will be different ten years from now. Ten years ago, the educational technology community as a whole was not discussing contemporary topics such as managing competencies and learning outcomes, interactive annotations, analytics, or connecting educational goals to labor market information. The discussion then was primarily focused on integrating the relatively new LMS into our university enterprises. If we projected forward only based on the trends from ten years ago, then ten years from now we might expect to have bigger LMS systems. What we consider to be the key issues today are not simple extrapolations of trends from the past.

Perhaps the question is not so much *what do the trends tell us about what educational technology will become* but rather *how do we make it easier to develop educational technology based on problems and paradigms not on our radar today?*

The Problem With an All-in-One System Approach

Online learning institutions like UOC (Universitat Oberta de Catalunya)[1] have been at the forefront in understanding that a single LMS neither fulfills all their educational needs nor is easy to adapt and evolve using one's own resources[2]. Changing a system like an LMS can be a difficult task. This is true for any enterprise-wide system, and current MOOC platforms fit into the same category.

A traditional, standalone LMS needs data from outside the system, such as the courses which are offered, the registered students and who the instructors are. Many organizations use a data feed approach for integration. The information is managed in one place, then imported into the database of the LMS. There are a number of issues with this approach. The data quickly becomes stale, can be difficult to reconcile during updates, and each system applies its own authorization techniques and rules which often do not align with the authorization or privacy policies of the university.

Incorporating information from multiple sources currently requires pooling all the data into a central repository and feeding it back out. When the number of data feeds is small and the scope of this data is confined to a single institution, this may not seem very complex. However, as the number of systems grows and as the scope of this data expands outside the walls of the university, then this becomes more of a problem.

Looking at the future of technology-enabled education, institutions will need to manage and integrate more and more information, such as learning outcomes, assessments, educational content, student performance, analytics data, job market trends, etc. This information needs to be accessible to a growing number of applications and systems throughout the educational enterprise and beyond. Access must happen in real-time, and include the ability to federate multiple information providers together without requiring intermediate data repositories. These trends are quickly outgrowing the traditional data exchange and synchronization approach to integration. They require us to move towards a more real-time integration approach supported by a *cloud of services.* The service providers are both internal and external to the institution.

1. UOC (Universitat Oberta de Catalunya). http://www.uoc.edu
2. See reference: Lowendahl JM (2011) Gartner Case Study. Approaching the Learning Stack: The Third-Generation LMS at Universitat Oberta de Catalunya.

Two Perspectives on the Problem

From one perspective, we are trying to better understand a new educational vision and new business models supporting that vision. Work from this viewpoint examines the role of technology in support of educational methodology from both the instructor and student perspective taking into account issues like mobility, interaction, and customization. It also looks at the current state of learning pedagogy while also understanding existing behaviors.

This work should produce recommendations for ways in which students and faculty can interact, ways in which students can plan their academic and professional careers, and ways in which education can be more effective.

But we do not want to create yet another single monolithic system to support new business models and methodologies nor do we want a set of disconnected applications and systems that do not integrate well with the enterprise. The architectural perspective is concerned with these integration problems.

The question is *to what degree does one perspective on the problem inform the other?*

There is not a simple answer to this question. If service modeling were solely dependent on an educational vision, then development investment would be at risk when the vision changes. An educational vision solely based on existing service models may not hold up to new ideas and concepts. It is a *chicken and egg* dilemma.

While these two perspectives often attract people from different disciplines, each has to complement the other. The perspectives differ in an important respect. Functional visions describe concepts for the purpose of achieving a goal. Service modeling organizes concepts to clarify integration of software components. Marching toward a functional vision without an integration strategy leads to monolithic systems, limits the ability to use third-party systems, and evolve in response to changing needs. This is at the heart of most misunderstandings between these two areas of work.

Service Architecture

Architecture, as it pertains to a service oriented approach, is concerned with service models. A service model does not describe a user experience nor does it prescribe a single methodology for teaching and learning. The purpose of a service model is to describe the integration points among the software components that will constitute such a system.

This may sound like an abstract technical distinction but it is key to designing sustainable systems. If we assume a multitude of methodologies and different styles of user experiences, all of which will continue to evolve and grow over time, then an architectural design cannot be constrained to a single way of doing things.

For example, a building may have been constructed to fulfill the needs of a factory. When the factory closes, the building may be repurposed as an office building, residential building, hotel, or a shopping mall. The basic needs of all these purposes are similar. The building should support its weight and integration with the electrical, water, telecommunications, and sewage utilities of the city. The degree of needed renovation and what can be preserved depends in large part on the design assumptions made for the original building. When the cost of renovation exceeds the cost of replacement, the building is demolished.

Some buildings are architected to anticipate radical change. They are modularized in such a way that parts of the building can be repurposed without affecting other parts of the building. The major concerns for designers are the integration among these building components, the ability to arrange these components in various ways, the ability to swap out these components, and the factoring of common services provided both from within the building and from the municipality.

A good service-based software architecture re-factors a traditional system into a set of reusable components that can be arranged and combined in different ways to achieve different goals.

This allows the system to better respond to changing needs without having to abandon the entire investment. Similar to the factory example, the degree of renovation will largely depend on the number of assumptions made at the outset. Strong consideration to integration, composition, and replacement will create a flexible system and maximize preservation of investment.

The view of a service-based architecture is different than a functional application view. The architectural view does not directly address how something will be used because it assumes that these usages will change over time. The architectural view is concerned with identifying underlying concepts and how these concepts relate to each other. These concepts tend to be more stable over time and thus provide a stronger foundation for continuous innovation.

For example, managing and applying *learning outcomes* is still a relatively new addition to the field of online learning. A functional view may prescribe tagging courses and videos with learning outcomes. An architectural view is that a learning outcome is an identifiable statement of a goal that can be achieved by performing some *activity* such as reading, watching, playing, or taking a course. Some of these activities may be informal while other activities are designed into a structured curriculum. The service-based architecture makes this distinction because the logistics of offered curriculum is fundamentally different than actions people can do on their own time. The service-based architecture may go on to model how these learning outcomes are used to describe job qualifications as part of a job market, company human resourcing, or business process workflow.

A good service-based architecture understands the relationships among these concepts without cemented definitions for things like *activities*, *courses*, *assets*, *jobs*, *processes*, or *learning outcomes*. Just like how a *space* in our example building may serve as an office, bedroom, or utility room during its lifetime. The trick is knowing where to be abstract and where not to be. The service-based architecture is primarily concerned with the interactions among software components to define stable specifications around their external behavior. This does not require advance knowledge of all the detail of its internal description. This is a design approach based on integration. The ability to modify these components within the integration framework is called *interoperability*.

A more complete building model would include the roads, highways, and trains that connect the buildings and nearby towns. Models wander outside the immediate problem space because it is the job of the architect to design how everything will fit together before it is built. The details of the implementation can be tackled later in the building process. The purpose of the service architecture is to tell us where we are, what services are needed, and where we can go from there. It does not prescribe what a user will see or how they will interact with the software.

Disaggregating the “LMS”

The Educause Learning Initiative publication *The Next Generation Digital Learning Environment: A Report On Research*[3]:

> *“Over time, the LMS needs to be supplemented (and perhaps later replaced) by a new digital architecture and components for learning that contribute to and enable the transitions that higher education is currently experiencing.”*

A vision of a next generation learning environment, where more and more of the educational experiences are being provided through a marketplace of applications, and where enterprise services are available to support such applications, begs the question *“what is left to do that we need a traditional LMS for?”*

One critical role of current LMS products in higher education is course administration from the instructor perspective. The LMS continues to serve a vital function at many of our institutions managing the administrative aspects of a course because these services are not otherwise available. This includes roles (who can author, take, teach, mentor, etc), sectioning, gradebook functionality and analytics and reporting functionality. These functions have little to do with learning approaches, but were put in place to support the administrative aspects of online education.

Factoring out this, and the other foundational services required to support effective and flexible online teaching and learning, leads to a simple sketch of next generation learning environments within an educational enterprise. These new environments will focus on “infrastructure” more than “platform” and will be comprised of two basic kinds of software components, *Enterprise Infrastructure Services* and *Educational Applications*:

Enterprise Infrastructure Services (EIS): These services support the next generation of institutional software infrastructure for education. EIS is the software that provides the foundational functional elements to support the application needs in the enterprise, and is based on institution-wide education and administrative business models. Today, these functional areas are well understood, and include assessment management, content management, learning outcome management, course management, to name just a few. More complete lists and specifications have been documented over the past decade and are ready to be leveraged for defining EIS. We will use the following graphical element to denote a functional service suite that may make up such an infrastructure:

In a marketplace reflecting a next generation learning environment vision, a particular institution could choose to have individual best-of-breed products covering each of the services that it desires to support, or elect to choose one product that covers them all. Many current LMS products are already strategically positioned to fill this market need. Access to the functionality of EIS is achieved through common *service contract* specifications (like those defined by the Open Service Interface Definitions (OSIDs)[4], or the Kuali Student/EagleApps Service Contracts.

The domain of education applies across the enterprise. The concerns of *training*, *self-help*, *assessment*, *results tracking*, and *digital content management* apply to campus personnel management in addition to their academic uses. Conversely, learning applications depend on information typically relegated to Student Information Systems, such as enrollment and grade submission. The concept of academic records are expanding to include student achievements and learning outcomes accomplished outside formal curriculum. All systems have dependencies on services which manage authorization, workflow, configuration, and calendaring, for example.

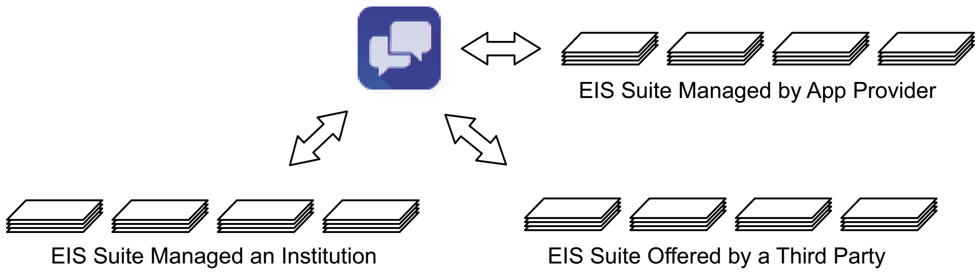
3. See Brown M, Dehoney J and Millichap N (2015).
4. See Appendix 2. The Open Service Interface Definitions (OSIDs).

When broken down in this way, a service applies to multiple business domains. It is the composition of the services and presentation of a user interface which deliver a business-specific function. A service contract is stable when it can be applied across domains without change.

The enterprise infrastructure is essentially a suite of well-defined services. Whether their implementations are hosted externally, hosted centrally, or some services are replicated for local consumption is immaterial because a true service environment hides the nature of their implementations as well as who is hosting these services. The federation technique can be used to combine a set of services implemented using the same service contract to create a single view.

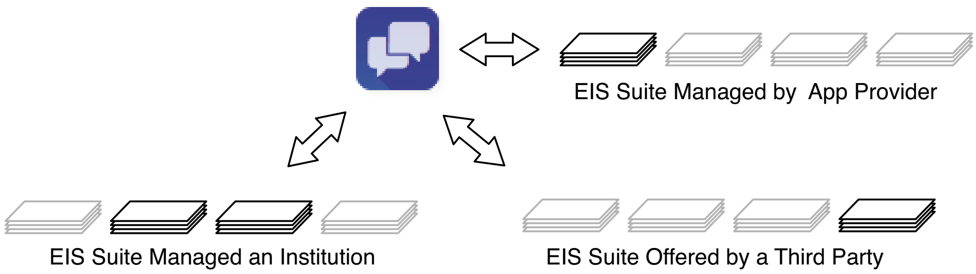
Educational Applications (Apps): For educators and students, and anyone else involved in the process of teaching and learning, the user experience we currently think about as the LMS can be served through a marketplace of apps[5], configurable to utilize the particular service endpoints supported by an institution. Put another way, this means that these applications are not always expected to persist their own data, and should be configurable to work with a particular institutional environment to utilize that environment's preferred service endpoints for various kinds of data persistence.

This doesn't mean that these applications should not be able to persist their own data, and to be viable products they will likely need to rely on their own default services if none are explicitly configured or if the institutional service provider does not support required functionality. The following graphic depicts a *discussion forum* application with three different service suites it can be configured to talk to.



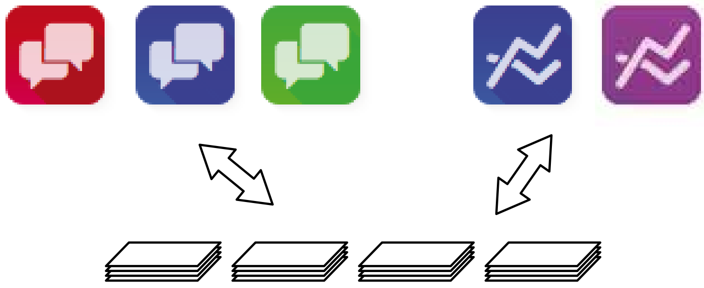
It be noted that there are many kinds of applications that, from the perspective of teaching and learning, can really just be considered as learning resources. A Javascript simulation of planetary orbits, or MIT's Mathlets[6], are examples of apps that exist more-or-less as interactive content elements, and do not need to be integrated with services in an educational enterprise to achieve their goal. At some point, however, the author of one of these resource apps may decide to add assessment functionality or to log analytic data, at which point it may be necessary to integrate with campus services for managing assessments, recording grades or submitting log entries.

There is also no reason that an institution wishing to make use of an application couldn't configure it to talk to different service providers for different functionality:



5. EduAppCenter - see reference - is a good example of a marketplace of apps in education.
6. MIT Mathlets Project. http://mathlets.org/

No matter where the suite of infrastructure services resides, or who manages them, the benefit will be to free the user functionality typically identified with the LMS from the stovepipe systems to which they are typically bound, and integrate it with the educational enterprise in meaningful ways. We will then be able to support educational software environments that allow for increasing application choice for faculty, students and staff, integrated through a common EIS suite. The following image illustrates three different discussion forum applications and two different *analytics tools* integrated through sharing the same enterprise infrastructure service suite.



Due to the evolving and experimental nature of educational software in particular, we can predict that the marketplace of apps will remain diverse and dynamic for the foreseeable future. This is in fact why specifications like IMS LTI[7] or Open edX xBlocks[8] exist, to assure that clever developers and motivated educators can freely extend and enhance their core learning environments with new methodologies and technologies.

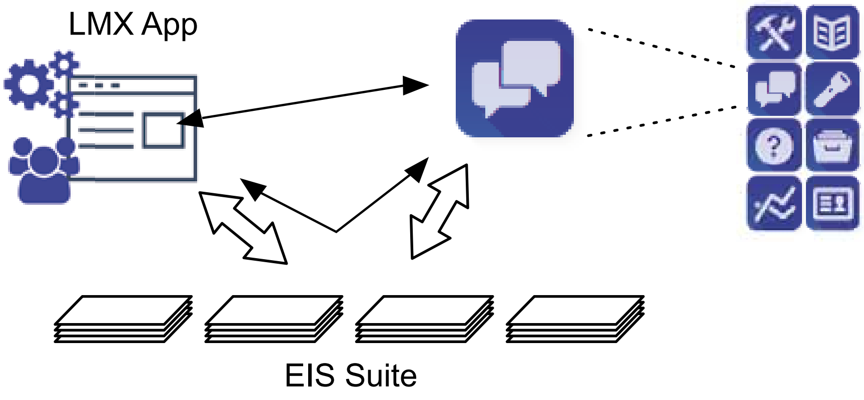
The "LMX"

Generically, we can imagine "classes" of applications that exist in this marketplace, defined either formally or informally, like the *discussion forum* application and *analytics tools* used in the previous examples. Other classes of applications that have emerged in the current marketplace include things like assessment (to test student knowledge in some new way), content management and discovery, simulation, collaborative authoring, document annotation, etc.

However, to complete our sketch of next generation learning environments, a new class of applications will need to be defined. Specifically these applications will provide the context and overall user experience required for a particular educational methodology or pedagogical model. Currently this kind of user experience is dictated by the legacy LMS platform itself. For instance, *edX* primarily supports a sequenced approach to learning, with students encouraged to move linearly through content and assessments. Platforms that support a competency-based approach may provide a very different overall user experience than those aimed at supporting an adaptive learning methodology. In the current LMS world, the choice of a platform more often than not restricts or at the very least strongly directs the educational methodology for its users.

By factoring out the core functional elements like course management as described earlier, and exposing those functionalities as services to be consumed by third-party applications, we can now think of a new class of educational application for authoring and managing the user experience for a particular learning methodology. Let's call this new class of application a *Learning Method eXperience (LMX)*.

7. See IMS LTI specifications.
8. See xBlocks. A EdX extensible mechanisms.



As is shown in the figure above, an LMX application utilizes appropriate underlying services to manage the educational experience for the learning methodology or methodologies it supports, and it can be configured to know where to find its service endpoints. It also serves as a UI container for, and can in turn configure other apps that will be part of the learning experience, similar to the kinds of apps that currently make up the IMS LTI marketplace[9].

This last part is important to stress: LMX apps are able to orchestrate and “launch” other educational applications as needed, and they can configure these educational applications to point to the service endpoints that the student, teacher or institution wishes to take advantage of, as appropriate. For web applications, IMS LTI 2.0 has begun to support this kind of ability to launch an application provider with service endpoint information.

9. See EduAppCenter as an example.

Modeling Learning Methodologies

As illustrated above, developing application functionality based on enterprise services becomes a much simpler task than developing an entire LMS. The complexity in developing a stand-alone system such as an LMS is largely due to replicating functionality such as authorization, content management, students, assessments, and calendaring. These services ought to exist in the enterprise cloud. A learning system should focus on using these services to implement learning methodologies without attempting to include the rest of the enterprise in a box. This is key to the architectural thinking for the next generation educational technology and requires a shift in the development process.

The following sections provide a sketch of such a process. The stakeholders in this process include educators, instructional designers, and potentially students, facilitated not by individuals with a technology focus but by architects whose primary responsibility it is to help the stakeholders articulate how they envision technology supporting their teaching and learning.

Process Outline

The elements of a learning methodology mapping process are outlined here.

Step 1 - Describe the Learning Methodology

Describe the learning methodology specifying functions and processes for the participating user roles (instructor, student, teaching assistant, mentor, etc) ranging from how the overall learning experiences are authored, managed and delivered. This is described by educators and instructional designers to the service architects who need an understanding of the desired outcomes.

A service architect is one who is familiar with the models and how to apply them. They must be able to have meaningful conversations with the business while mapping their ideas and requirements to the service models. There must be an understanding in the process that a language translation is occurring. This is a two-way conversation as the business end is also informed from the service models.

This description should be free of any technological considerations and free of any user interface behavior.

Step 2 - Service Model Identification

Mature service models tested over time and applied to different scenarios are preferred to inventing a new service model for a specific situation. Mature service models are more interoperable and far less likely to require modification when the scenario changes.

Using a broad suite of service models like the OSIDs (see Appendix 2), identify which service domains (packages) are pertinent in supporting the desired learning methodology. The OSIDs are based on a collection of 180+ service models compiled and vetted over 15 years, and a particular vocabulary has been defined over the course of developing the models. It is unlikely that existing service model will use the same vocabulary developed in step 1. A goal of this process is to align these vocabularies with the help of a service architect.

This identification step is best done by aligning the entities. For example, let's take a learning methodology that refers to a *classroom* to represent a class with an instructor and students. Someone looking at this term out of context may think it is the physical room in which a class is taught.

In the OSID world, the term *classroom* does not exist, however there is a Room entity in the Room package. Looking at the associated Room model, it can be seen that a Room is part of a Floor which, in turn, is part of a Building. But this is not what the learning methodology meant when referring to *classroom*. The learning methodology is more likely concerned about instructors and students. Looking around for these relations throughout the models, we find a *CourseOffering* entity that includes instructors. CourseOffering is managed as part of the broader Course package. Further investigation will reveal other entities in this package required to map the idea of classroom as described in the learning methodology.

Gaps remain between the methodology vocabulary and the model, but for now it suffices to add the Course service to our inventory of service packages, or *service domains*, required to support the learning methodology.

This is a step performed by service architects who understand the service models. This is an iterative process with continued conversations with educators and instructional designers. Examination of these models will also help inform the description of the learning methodology which may tangle or omit important concepts.

Another common problem is dealing with functional descriptions derived from existing applications. These tend to articulate functional requirements in terms of how those applications operate. An example is a legacy application which displays a learning outcome along with its attribute of *next* learning objective to help the student navigate. The Learning OSID model, however, says that there are many possible learning pathways that can reuse specific learning objectives. According to the Learning OSID, the ultimate goal needs to be specified in order to understand the next set of possible learning objectives that will get the student there. This was incompatible with the existing application but, in this case, the OSID model described the way faculty wished their application worked in the first place.

While we are describing this process in the form of steps, an effective process is rarely linear.

Step 3 - Functional Application Breakdown

This step describes the set of user functions that need to be performed to fulfill the learning methodology in light of the identified services. For example, learning objectives may be included in a learning methodology description. We will know from step 2 that the Learning service defines an Objective as part of its model. Next is to define the kinds of applications that will manage and manipulate the learning objective.

Applications could be selected from a marketplace or developed from scratch and in either case integrated through their specific APIs[10]. Then, the first level of this breakdown may simply identify, from these applications, categories of functions such as management of learning objectives. It's important at an application level to separate this from other kinds of functions such as searching for courses or authoring of assessments.

The second level of this breakdown calls out specific functions on an entity by entity basis. Examples are viewing a learning objective, updating a learning objective, or rearranging the learning objective hierarchy.

While this can be done by analysts or UX designers, the service architect also plays a role in helping communicate the service models and performing the breakdown.

Step 4 - Service Model Mapping

The result of step 2 may have only reached the level of the service package. Here, functions identified in step 3 should be mapped down to the entity level. It may be the case that more entities exist than was expected.

For example, step 2 determined that a *classroom* in the learning methodology is named a CourseOffering in the OSID models. We found the instructor(s) we were looking for but did

10. Application Programming Interface (API).

not find the list of students. To solve this problem we examine the places (In perhaps multiple models) where CourseOffering is used.

One such place is the course registration model. The Course package defines a separate entity called CourseRegistration which serves to connect a student to a CourseOffering. It is not modeled as a list of students but a set of relationships. Why the indirection? Because it has been proven to be of general concern to know if the student is currently registered, had been registered at any point during the term, how they are registered (full-time, pass/fail, auditor), and whether they had dropped out. These concerns may not have arisen in the methodology discussions. However, when the methodology description makes its way to a real-world system, how the students will be graded, maintaining access to course materials and prior work, detecting schedule conflicts, and accurate reporting are some of the issues that will need to be addressed and already covered by this model.

This work also feeds back to step 3 to further complete the list of necessary functions and a service architect familiar with the models is best suited to help map and guide this process.

Step 5 - Identify Service Application Components

Mocking up and developing an entire screen often runs into conflict with the service models as application-specific APIs are developed to make that particular screen composition efficient. The means to assemble an entire view across many different services tends to dominate the effort to establish good service models. This has caused many projects to abandon the effort to establish common services. However, applications can be factored in a similar manner as the service models themselves. Their concerns need to be organized and modeled. It is not just a task to make data appear on a screen but achieving an understanding of what components make up an overall application.

An application or application element is composed of smaller independent application components. The smallest application components consume a single service. What we see on our screens is a composition of these components. This compositional style is helpful for accelerating development by keeping it focused on smaller reusable pieces. Mixing and matching of components can offer a variety of end-user experiences. The LMX is an example of a single application component or a small composition of application components.

The UX designer may be responsible for the specific styling of an application component. However, it is the service model that ultimately defines its scope. The UX designer is responsible for the composition of these application components into meaningful user experiences but within the service model framework.

Clearly this is not a sequential workflow. It is an iterative process that involves diving into detail and popping up to grasp the bigger picture. The goal is an architectural blueprint that will drive the construction of applications and services or integration with third-party components available in a marketplace. This blueprint is formed both by the needs of the business and the integration requirements satisfied by the service interface definitions. It does not concern itself with technical detail, place a reliance on common software platform, nor define system boundaries. The result is a more accurate and complete functional description as well as a clear understanding of the many integration points around which development activities can occur.

The process is applicable to both newly developed software and integration of existing software available in the marketplaces of application and service components. Integration without locking into a single technology platform requires a decoupling of the various service and application components. Breaking down and aligning functions along the service interface definitions will ensure this decoupling.

This process places the services at the heart of the workflow with the role of a services architect binding both the functional and technical concerns at all levels of the project. This is a new kind of role that should not be relegated as solely a technical concern as it moves across all areas of a project. It relies on stable comprehensive modeling to guide functional thinking and minimize the impact of technical design constraints from curbing desired outcomes.

Example - Flipped Classroom Methodology

To begin to illustrate this process, we will use an example of what is commonly referred to as the *Flipped Classroom* teaching and learning methodology. A deeper analysis of this learning methodology is available in the appendices, and a quick scan will provide a good sense of the kind of thinking that goes into developing a mapping between a methodology and the applications and services required to support it.

Description of Methodology

A computational aerodynamics class meets twice a week, 1.5 hours per in-class session. Prior to each class, students are expected to read the materials and view short videos covering the topics for the next class session. After reading and viewing, students take a brief conceptual assessment to test their basic understanding of the concepts presented. This homework activity is meant to replace the traditional in-class lecture. There is a date and time, prior to class time, by when all students are expected to have completed the flipped homework assignment.

Prior to class, the professor reviews the results of the conceptual assessment and decides whether to take 10 minutes at the beginning of the in-class session to review particular “fuzzy” topics, or perhaps identify students who may need some extra attention during in-class activity.

During in-class time the students are split into groups of two or three (perhaps organized by shared mis-understandings based on the assessment). Together they work on the problem set for that class session, which they are expected to finish by the end of the class period. The professor and TAs keep an eye on everyone to offer help or to take a mini “lecture-break” to address any common issues they observe.

Overall Application and Service Mappings

A number of primary service packages will be utilized as part of the EIS suite to support the functionality and data persistence of these applications (a complete set of defined services packages is listed in the appendix):

> *Learning Service Package* - Author, map and manage the learning outcomes for the course. Conceptual assessments, problems in problem sets, and elements of the Syllabus can be related to one or more individual learning outcomes.

> *Repository Service Package* - Author, manage, and deliver the learning assets that the students will experience as part of the homework.

> *Assessment Service Package* - Author the conceptual problems, arranging them into assessments, deliver those assessments and track student responses. Author and manage problems and problem sets to be printed and handed out in class.

> *Course Syllabus Service Package* - Define the overall structure of the course, authoring the “flipped” homework assignments, grouping them into modules.

> *Course Plan Service Package* - Manage a particular offering of the syllabus in a particular semester to determine how the structure of the course as defined by the Syllabus service package maps to a particular schedule of the offered course.

> *Messaging Service Package* - Send and receive messages between users utilizing preferred messaging technologies. In particular for messaging students about information they should know prior to the in-class activity based on flipped homework assignments.

> *Grading Service Package* - Apply and manage grades or ratings.

A exercise to identify application elements and components required to support the flipped classroom methodology can then be facilitated. The result of this exercise is a list of application elements required to support the methodology:

> *Learning Outcome Authoring Application* for creating the inventory of course related learning outcomes that will relate the content, conceptual questions and problems for each homework assignment and problem set.

> *Assessment Authoring and Management Application f*or Authoring, cataloging and assembling problems for inclusion in problem sets and conceptual questions for the flipped homework assignments.

> *Curricular Asset Authoring and Management Application* for authoring and cataloging the content (videos, PDFs, etc) to be delivered as part of the flipped homework assignments.

> *Homework Activity Authoring Application* for designing the student experience, bringing together content and conceptual assessments for the homework assignments.

> *Student Learning Method Experience (LMX) Application* for managing the overall student experience for the flipped homework assignments.

> *Instructor Assessment Dashboard Application* for reviewing students' conceptual grasp of the content in preparation for the in-class activity.

> *Grade Sheet Application* for entering problem set grades and reviewing trends.

These application elements can in turn be further refined into discrete components that define the user experience and provide a detailed mapping to underlying service elements. In this document we will not go into this detail for the example methodology, but a successful process would require it.

Other Methodologies

Using this same basic process we should be able to tackle any learning methodology for which we wish to develop or assemble a system. Wikipedia has a pretty complete list that is a good starting place for consideration[11].

Some learning methods that stand out as high priority for modelling and application/service development in higher education include:

> Collaborative Learning: Students work together to achieve the same common product. Students are presented a complex case study. Diversity contributes to the learning process.

> Project Based Learning: Students achieve a set of learning outcomes by engaging in an extended project or challenge, usually of some level of complexity and requiring significant time.

> Case Based Learning: Students are presented with cases, either real/historical or simulated, and are expected to put themselves in the role of people dealing with that situation. The final artifact may be a solution to the case, or other evidence that the student has considered the case in depth.

> Adaptive Learning: The results of student activity, assignments and/or assessments are used to adapt the ongoing presentation of materials and assessments towards an educational goal.

> Competency Based Learning: Student activity, assignments, assessments, projects, etc. are designed around achieving proficiency in one or more competencies or learning goals. These competencies may or may not be organized into pathways or hierarchies that help drive the learning experience.

11. Learning Methods. Wikipedia. https://en.wikipedia.org/wiki/Category:Learning_methods

In real educational settings we know that learning methodologies such as these are often combined, and in mapping to applications and services we should also be able to use applications and learning method experiences together to achieve particular functionality. For instance, to support a competency based methodology that utilizes case-based learning approaches. This is an area where the LMX approach, allowing multiple different LMX applications to interoperate or one LMX to support a set of methodologies, provides more flexibility than the traditional LMS.

Benefits of This Approach

By identifying and refactoring the last vestiges of functionality that legacy LMS's currently provide, namely basic course management services and the methodology-driven learning experience (UI/UX), we can extend a vision of next generation digital learning environments consisting of a marketplace of Enterprise Infrastructure Services and a marketplace of educational applications, of various types or classes, that consume Enterprise Infrastructure Services.

The Learning Application Marketplace

It is expected that a growing and competitive marketplace of educational applications will be available to support consumer choice. "App stores" will likely provide a vehicle for promoting, discovering and acquiring these applications. Service contract specification will become the glue allowing these applications to work together effectively and integrate with the educational enterprise.

This marketplace becomes even richer through the identification of a class of Learning Method Experience (LMX) applications that orchestrate the user experience in support of different kinds of learning methodologies, and providing user interface containers for hosting other educational applications and resources.

The Enterprise Infrastructure Service (EIS) Cloud

The data underlying these applications will be well managed as infrastructure services so that transitioning from one application to another will not involve replicating or transferring data. Educational enterprises can mix and match between "service providers" as required in building out their infrastructure service suites, which can include services hosted locally at an institution, or externally hosted services hosted.

Additionally, service infrastructure endpoints can be federated to provide educational applications access to functionality across implementation technologies. A campus learning object repository, a library system, and a campus Google Drive might be exposed through a single service contract. This technique can also be used to share information among and between institutions, for instance, to share a learning outcome tree or competency map that spans schools.

The service *cloud* is the suite of services supporting multiple business domains using the same service contracts. This cloud extends from the local installation through the central enterprise through an array of potential external service providers. The goal of a stable service contract is to be used in any or all service provider scenarios and across multiple business domains without ever having to touch the code in the application.

In that sense, Enterprise Infrastructure Services, components, and third-party cloud services implemented against the same service contracts can be exchanged in marketplaces in a similar manner to present-day appstores.

Continuous Evolution

The effort involved in enhancing an LMX application to incorporate some new feature related to evolving learning methodology will be relatively small, with incremental change incurring only incremental cost. An example of incremental change may be adding the ability to layer learning outcome pathways on the content being developed for a course. With current systems, this kind of incremental change typically requires substantial effort to implement. The goal is to promote educational ecosystems of continuous change and advancement in our educational tools as well as their technological dependencies, rather than infrequent leaps forward.

In addition, the entire architectural model promotes new approaches to the process of designing and developing/integrating educational applications, rooted in learning methodology as opposed to mired in technology. Building LMX and other classes of applications upon a composition of lightweight components (the heavy lifting being delegated to underlying infrastructure services) will ease technical transitions from one method-driven set of features to another.

Improved Development Process

The process describes a way to articulate functional requirements in terms of existing service models/interfaces and using this as the blueprint for building and sharing applications and service providers. This document did not describe the efforts required in building and vetting mature service models as the two major service frameworks listed above have evolved over many years. Nonetheless, the process adds an additional role and service mapping work not found in development processes where functional requirements are fed directly to programmers.

Functional requirements tend to be sequenced in one of two ways:

Incrementing from simple to complex
The problem with this approach is that what is deemed simple and what is deemed complex is based on (1) an application-only perspective and (2) a preconceived notion of how the technical implementation is designed where such a design is based on the simplified requirements. Often, a project gets stuck when the complex requirements arrive resulting in a major or total replacement.

> For example, in designing an online learning assessment system it may have been specified to develop a low-stakes assessment prior to a high-stakes assessment. The low-stakes application will directly access the questions and answers with no tracking of responses. The high-stakes application will submit each response and takes its cue from the service. However, in moving between the two scenarios the backend, protocol, and application design needs to significantly change to the point of building much of it twice. The driver in the move may not just be the security of the assessment but the tracking, analytics, adaptive questioning, and the ability to identify needed learning objectives to assist the student in their learning. The low-stakes assessment is even less useful than it might have seemed.
>
> Incremental development is good when it continues to move forward. It is not good when it results in a growing set of things to rework with each new desired functional outcome. Having models of the end-game are critical to avoiding the one step forward/one step backward problem.
>
> **In order of business value**
> Business value to the sponsors is very important to understand. However, it must be understood the work breakdown and needed infrastructure before jumping into development. In reality, temporary measures are performed to demonstrate business value while others work on more permanent solutions that take into account the total cost of ownership. This should be done in a contained manner that does not ripple throughout the entire system. Use of the services as boundaries to contain these ripple effects are very effective containers.
>
> Again, it's also important to use the service models as a guide in breaking down functional requirements. They are broken down enough when a developer's activity is focused on one single service or application component at a time. When developers slice through a system by hopping around, the parts that ought to be decoupled and composable are instead glued together and cannot be reused in different contexts. This is the chief problem with most platforms.

Agile development methodologies do not prescribe architecture. Unfortunately, getting stuck on a technology-based platform tends to reduce velocity over time because there's less and less that can be done with it without unravelling the entire thing. However, the kind of architecture that can be achieved if based on well-factored service models rather than technology stacks is one where the same ongoing effort builds bigger and bigger things. The models inform the construction of service and application components. These components are put together to make even bigger components, and so on. Although there is more planning work up front, the effect is an increase in velocity over time from the ability to move larger pieces around.

This is possible because of the emphasis on using interfaces to describe *all* integration points based on complete service models. The goal is to get on with building larger and richer learning environments than was possible before.

Current Initiatives and Next Steps

To achieve this ultimate vision of next generation digital learning environments, it is important to begin exercising the processes, architectural practices and application and service modelling outlined in this document.

Current Initiatives

As of this writing, there are a number of development projects in higher education that are taking this approach:

> *Fly-by-Wire (FbW)*[12]: Led by MIT professor Karen Willcox and Associate Dean of Digital Learning, Vijay Kumar, is leveraging assessment and learning outcome services to improve student achievement by enabling teachers to provide differentiated instruction in a scalable way. This project is testing the theory and that student achievement can be scalably, significantly increased through (1) combining tested methods of modularity and curriculum mapping to create competency-based mappings, (2) creating a companion “Fly-by-Wire” technology that enables teachers to differentially guide students towards competencies, and (3) deploying this approach on a flexible framework that integrates open technologies to achieve cost-effective scalability. MIT has partnered with community colleges in Massachusetts and Colorado to test the FbW system. FbW is being funded through a First In The World grant from the U.S. Department of Education and is being built upon the OSID specifications.

> *The Woodrow Wilson Academy CBE Platform*[13]: The Woodrow Wilson Academy (WWA) opens its doors in June 2017 to its first matriculating class. At that time WWA will have a learning environment in place that can support key elements of its innovative competency based education (CBE) methodology, including its learning outcome models, cross-cutting challenges, activities, placement tests, assignment management, resource libraries, portfolios, etc. In addition, in the year and a half preceding the initial class entry, the faculty and staff of WWA will need to have tools in place for modeling competencies and learning outcome, authoring assessments and managing content, and will run experiments related to various elements of the educational methodology. The WWA is being funded through the Woodrow Wilson Foundation and OSIDs are being used to inform this modelling of the project.

> *The Connected Learning Initiative (CLIx)*[14]: CLIx is a bold and innovative effort to improve the professional and academic prospects of high school students from underserved communities in India. CLIx incorporates thoughtful pedagogical design and leverages contemporary technology, including online capabilities, to provide quality educational content and experiences at scale. The initiative aims to reach a total of approximately 1,000 schools and 150,000 students in 4 states during 2015–2017, as well as conduct professional development for approximately 2,700 teachers. Given the scale of the project and diversity of learning environments, CLIx will need to take a multi-platform approach. A set of learning services and related applications are being enhanced through the CLIx project to help meet content authoring and delivery across multiple target platforms. and OSIDs are being used to inform the modelling of the project. OSIDs are being used to inform the modelling of this project.

> *Outcome Exploration System (Xoces)*[15]*:* The Xoces project is documenting, organizing, and connecting learning outcomes across the undergraduate curriculum at MIT and SUTD (the Singapore University of Technology and Design). At MIT, it has collected outcomes for the

12. The Fly-by-Wire Project. http://fbw.mit.edu/
13. Woodrow Wilson Academy. http://www.woodrowacademy.org/
14. CLiX Connected Learning Initiative. http://clix.tiss.edu
15. Xoces Outcome Exploration System. http://xoces.mit.edu

undergraduate curriculum in the Department of Aeronautics and Astronautics, but is also moving into documenting outcomes across other disciplines. With SUTD, the entire four-year undergraduate curriculum has been mapped. The project has been developing effective ways to display the information graphically to support curricular and program planning. All learning outcome information is being managed through enterprise infrastructure services. Xoces is funded by the Lord Foundation, MIT's Office of Digital Learning and Singapore University of Technology and Design's (SUTD) International Design Center. Xoces is built upon OSIDs.

- *New UOC online degrees on Design and Arts*: The Open University of Catalonia (UOC) has been designing new degrees on Design and Arts. As a fully online university, UOC is committed to implement new learning methodologies and UI/UX approaches that foment the creativity and visual aspects as well as provide resources and tools to work the required competencies and skills online. The LMX App approach feeds perfectly with this new requirements.

- *Competency-based learning at UOC*: The Bologna Process is a series meetings and agreements between European countries designed to ensure comparability in the standards and quality of higher education qualifications. This process has had many knock-on effects such as bilateral agreements between countries and institutions which recognise each other's degrees. At the same time, the Process is also moving away from a strict convergence in terms of time spent on qualifications, towards a competency-based system. The Open University of Catalonia (UOC) has a set of specific and transversal competencies linked to each programme and subject and has been working in a project - using OSIDs - to map competencies at a learning activity level with the aim to map and visualize programmes in terms of competencies and to establish relation between competencies, subjects, learning activities and labour market.

- *UOC Learning Platform*: The Open University of Catalonia (UOC) has been working with the team in MIT's ODL/SEI for close to a decade to help define, test and refine the service contract definitions that Backstage is built on. UOCs current platform, Campus 5.0, was built in 2008 using a number of these service contracts including OSIDs. Now, UOC is planning an aggressive new project to develop its next generation platform which they intend to deliver within 4 years. This document is itself part of the effort to define the roadmap and approach for UOC next generation platform.

- *Tsugi Project* (Charles Severance)[16]: Tsugi is a framework for developing Learning Tools that involves IMS standards. Its goal is to build a scalable multi-tenant "tool" hosting environment based on the emerging IMS standards to help move the industry toward a Next Generation Digital Learning Environment. Current environments can provision and launch those tools using a common "App Store" (marketplace of apps).

Projects like these are pushing forward the vision of next generation digital learning environments, and provide open-source reference service implementations and applications. They are utilizing many, if not all, of the design processes that have been discussed, including in the cases of the WWA, FbW and UOC projects, modelling innovative learning methodologies envisioned by faculty and educational thought leaders.

Next Steps

Moving from these handful of projects to industry-wide advancement and change will require additional investment and prioritization of existing investment in educational technology projects. Organizations that fund initiatives like these are encouraged to help make opportunities to:

- Initiate projects aimed at testing and advancing this design approach and process

- Initiate projects focused on delivering and mapping innovative learning methodologies and pedagogical approaches

16. Tsugi Project. http://tsugi.org

> Promote the development, use and advancement of next generation service-based integration specifications and techniques.

> Work with industry organizations (both open source and commercial) to promote and develop service and application marketplace models

Ultimately, it is the responsibility of consumers of educational technology - teachers, student, instructional designers, institutional decision makers, etc. - to promote their desires and requirements for next generation learning environments. Existing consumer focused organizations, like Educause/ELI, or the regional educational technology consortia that exist in many states, nations or across educational systems, are uniquely positioned to help organize and amplify the needs of members directly engaged in teaching and learning.

Acknowledgments

Special thanks to Malcolm Brown, Charles Severance and Marc Alier for contributing to discussions at the 2015 Barcelona workshop, where a number of ideas in this document were refined. To all people from MIT and UOC teams that were involved in some way in that work. Their comments, reflexions and reviews constitute the fundamentals of the document.

We gratefully acknowledge the contributions of Scott Thorne over the years forming the basis of the thinking behind this document with his pioneering efforts in creating the OSIDs and Kuali Student services architecture.

References

Alier M, Mayol E, Casañ M, Piguillem J, Merriman J, Conde M, García-Peñalvo F, Tebben W and Severance C (2012) "Clustering Projects for eLearning Interoperability", Journal of Universal Computer Science, Volume 18/Issue 1, 2012.

Barber M, Donnelly K and Rizvi S (2013) "AN AVALANCHE IS COMING. Higher education and the revolution ahead" Institute for Public Policy Research. http://www.ippr.org/publications/an-avalanche-is-coming-higher-education-and-the-revolution-ahead

Brown M, Dehoney J and Millichap N (2015) "The Next Generation Digital Learning Environment: A Report On Research" EDUCAUSE. ELI Paper. http://www.educause.edu/library/resources/next-generation-digital-learning-environment-report-research

CLiX Connected Learning Initiative. http://clix.tiss.edu

EduAppCenter. An open LTI app collection. https://www.eduappcenter.com/

IMS LTI. IMS Learning Tools Interoperability specification. IMS Global Consortium. https://www.imsglobal.org/activity/learning-tools-interoperability

IT Service Management. Wikipedia. https://en.wikipedia.org/wiki/IT_service_management

Kumar V and Merriman J (2005) "From Course Management Systems to Open Frameworks and Sustainable Ecology of Educational Systems" Course Management Systems for Learning: Beyond Accidental Pedagogy, IGI Global, 2005, 331-343, Print.

Learning Methods. Wikipedia. https://en.wikipedia.org/wiki/Category:Learning_methods.

Lowendahl JM (2011) "Case Study: Approaching the Learning Stack: The Third-Generation LMS at Universitat Oberta de Catalunya" GARTNER. https://www.gartner.com/doc/1607419/case-study-approaching-learning-stack

MIT Mathlets Project. http://mathlets.org/

NEW MEDIA CONSORTIUM and EDUCAUSE ELI (2015) "NMC Horizon Report > 2015 Higher Education Edition" NMC Horizon Project. http://www.nmc.org/publication/nmc-horizon-report-2015-higher-education-edition/

Open Service Interface Definitions (OSIDs). "About the OSIDs". http://wiki.okapia.net and specification webside: http://osid.org

Santanach F, Prieto J (2015) "El futuro de la educación: Como la tecnología puede impactar en la educación" Novática, 234 "Año 2025: El futuro de la Informática" http://www.ati.es/spip.php?article2817

The Fly-by-Wire Project. http://fbw.mit.edu/

Tsugi Project. Multi-tenant learning tool hosting platform. http://tsugi.org

Woodrow Wilson Academy. Woodrow Wilson National Fellowship Foundation. http://www.woodrowacademy.org/

xBlocks. Open EdX. EdX Inc. https://open.edx.org/xblocks

Xoces Outcome Exploration System. http://xoces.mit.edu

Appendices

Appendix 1. How Technology Can Impact Education[17]

Foreseeing the future is always a risky business. In this appendix we try to describe a possible world based on what is happening now in technology and education and the potential consequences that may arise from this situation. The educational process is a complex matter which requires observation and study from different disciplines. Here we will view it primarily from the technological dimension, which makes us leave to one side many other transformative realities which will also have an impact. Consequently, we once again situate ourselves in a risky business. However, assuming this risk, let's try to do it:

Computerisation has brought virtual teaching to education. Today there are many digital platforms which allow students to learn and teachers to teach anywhere, regardless of their physical location, and at any time. This has changed the role of the teacher: they move from the teaching classroom and become more of a mentor or coach. At the same time, the student must assume a more active role, questioning their classmates and requesting mentoring from their teacher. Over the next ten years, this change, which thus far has occurred in online education, will extend to the physical classroom with the application of what is known as the "flipped classroom", where the student will have the necessary technology to study theory at home before attending classes to do practical exercises, to generate debate and to be guided by the teacher. Thus, the methodologies based on problem solving or project work will have more weight; they will be necessary tools to stimulate creativity, to support the coaching dynamic and to encourage discussion among students. The speedy incorporation of technological competencies in all disciplines will mean that one of these tools for creativity will be programming, with languages such as "Scratch" in primary and secondary education and "Python" in further education. Agile methodologies will also have their opportunity and could become a good tool for project based learning and the promotion of interdisciplinarity.

The first computers occupied an entire room and required several people to operate them. At the time it was difficult to think of a computer as something separate from the industrial world or which was not used to perform corporate tasks, yet in just a few years the personal computer exceeded all expectations. The vision of the computer as a personal tool, which could assist in the jobs assigned to an individual in the workplace and also be used in the personal sphere, led to the democratisation of information technology and a complete transformation of many jobs in both their physical configuration (a desktop computer) and their skill dimension (digital literacy). This represents a profound change in industry and in society, the impact of which probably goes far beyond what was foreseen.

It therefore seems clear that, just as the shift from the agrarian to the industrial society made literacy essential, the shift to the digital society makes digital literacy essential. We understand digital literacy to mean the set of actions which makes it possible for people to be incorporated into the knowledge and activities developed around ICT (Information and Communications Technology).

However, as stated above, the impact of ICT on the workplace has transformed the majority of professions. Jobs are increasingly inter-related with the digital world given that the instruments for these professions are increasingly digitalised, as are the working and communication methods. Consequently, we now need to learn new ways of working and we need to use increasingly digitalised instruments which will continue to change and vary in format frequently throughout our working lives. Moreover, the very introduction of digital instruments into the professions accelerates innovation, because it signifies an injection of technology and communication which previously only came from heavy investment. Therefore, the knowledge and skills required to exercise these professions are acquired and have an influence throughout our professional lives. There are very few professions today which in practice do not require, apart from classic knowledge, constant and life-long recycling.

17. Summarized and rearranged from a Novatica press article: See Santanach F, Prieto J (2015).

This is a phenomenon leads us to a globalisation which fosters research and innovation as well as promoting exchange and interdisciplinarity, not to mention business internationalisation. Today professionals work in distributed, interdisciplinary and international teams, which make use of various disciplines. Most importantly, many of the instruments now used in the world of work did not exist or were experimental five or ten years ago. In practically all fields, innovation is continuous and recycling skills throughout life is an imperious and obligatory necessity.

Thus, university and professional training are no longer the final stage of education; they are now the first step in a lifelong path of education. We no longer talk of a doing a master’s or postgraduate course in order to evolve professionally or change professions; instead, lifelong training is essential to keep up to date in the same profession. Over the next few years, we will witness a formalisation of this lifelong training, which currently lacks official recognition. Although companies are presently the ones which have initiated this path with the so-called “corporate universities”, we believe that it is inevitable that educational institutions, particularly universities, will have to react and that they are the ones which are best equipped to fill the space for lifelong education. At universities, this transformation implies a redefinition of the concept of engineer or graduate. At present a university degree provides students with skills linked to their branch of learning which they may never use in their working life yet they do not acquire skills from other disciplines which they will need in their chosen field of work. Therefore, universities must somehow take advantage of the possibilities offered by lifelong learning, removing the pigeonholing and pre-configuration of each profession and allowing the profession to define itself based on the day-to-day reality of its activities. Knowledge, subjects and skills are universal and universities explore and nurture their growth each day, they are equipped to offer knowledge; doing so without pigeonholing it does not seem an unnatural transformation.

Likewise, the increasingly technical nature of professions brings the need for literacy in terms of technologies and their uses. At the same time, the users’ increased technological knowledge also further catalyses digitalisation.

In this accelerated spiral of digitalisation, we now find ourselves in the age of the mobile device. Smartphones and tablets—alongside the computer, which has also become mobile (with the laptop)—constitute the access points to this digital world. In education, these devices provide new spaces and new temporal availability for teaching and learning: while waiting to cross the road I can use my smartphone to see whether my teacher has marked my assignment or posted a new learning activity which I can do later on my laptop; or I can make use of my time on commuting time to study digital content which adapts well to my tablet. All of these are everyday situations in learning which mobile devices help us to tackle. They are also a synonym for efficiency; the infinity of apps we download for different purposes have one thing in common: they help us to do things better. If an app is frustratingly inefficient or not entertaining enough, we delete it. In the coming years, education faces the challenge of knowing how to choose from or add to this universe with applications which best adapt to the uses and activities of student and teacher. It would be well to view apps as allies which make the educational environment a rich ecosystem which can be personalised, without forgetting that it is a much more savage environment than the Internet: only the most efficient and user-friendly app survives on our device.

However, mobile devices are only the first reflection of the Internet of Things (IoT). In parallel to the consolidation of the mobile world, this new concept will gather strength in education. The possibilities of connecting and linking physical objects with the digital world are immense in the education sector and will lead us to a redefinition of what we understand by online learning. Indeed, we believe that in our digital society most of the education models will include online learning even in face to face situations.

These new uses for the online environment will progressively change the concepts of universities and professional training. Let us imagine that hospital equipment is connected and that residents can have their practical assignments monitored by teachers in other hospitals or in the medicine faculty itself. Let us imagine that work carried out in a museum and automatically digitalised is validated as a learning activity. Let us also imagine a connected factory in which the production line can be manipulated and monitored remotely for educational purposes. In conclusion, with all these new infrastructures we could educate and train from the best places to learn, including the workplace itself, and enable the managers of these facilities to capitalise on them even more.

Appendix 2. The Open Service Interface Definitions (OSIDs)[18]

The Open Service Interface Definitions (OSIDs) is a service-based architecture to promote software interoperability. The OSIDs are a large suite of interface contract specifications that describe the integration points among services and system components. The goal of the OSIDs is to remove platform dependence by fostering choice among a variety of different and independently developed applications and systems, allow independent evolution of software components within a complex system, and federate service providers across organizational and technological boundaries.

The OSIDs were initially developed in 2001 as part of the MIT Open Knowledge Initiative Project funded by the Andrew W. Mellon Foundation to provide an architecture for higher education learning systems. OSID 3K development began in 2006 to redesign the capabilities of the specifications to apply to a much broader range of service domains and integration challenges among both small and large-scale enterprise systems.

The following table lists the primary service domains defined by the OSID specification:

Package	Description
> Acknowledgement	The Acknowledgement OSID defines and relates credits such as authors or creators to OsidObjects.
> Assessment	The Assessment OSID provides the means to create, access, and take assessments. An Assessment may represent a quiz, survey, or other evaluation that includes assessment Items. The OSID defines methods to describe the flow of control and the relationships among the objects. Assessment Items are extensible objects to capture various types of questions, such as a multiple choice or an asset submission.
> Authentication	The Authentication OSID manages authenticated entities.
> Authorization	The Authorization OSID manages and queries authorizations.
> Bidding	The Bidding OSID provides a means for running and bidding in auctions.
> Billing	The Billing OSID manages billing for customers.
> Blogging	The Blogging OSID manages blogs.
> Calendaring	The Calendaring OSID manages events, commitments and calendars. The Calendaring OSID offers a rich set of event management services.
> Cataloging	The Cataloging OSID provides a means for organizing and federating OSID objects.
> Checklist	The Checklist OSID provides a means of managing action items.
> Commenting	The Commenting OSID provides a means of relating user comments and ratings to OSID Objects.
> Communication	The Communication OSID allows OSID Providers to communicate with OSID Consumers asynchronously.

18. See Reference: Open Service Interface Definitions (OSIDs).

Package	Description
> Configuration	The Configuration OSID defines a set of interfaces for configuration retrieval and management. The Configuration service may be used to store user profiles or as a means to assist OSID implementors modularize configuration data.
> Contact	The Contact OSID is an auxiliary service that relates contact information to OsidObjects.
> Control	The Control OSID manages control systems. A system is comprised o f Devices whose Controllers are the control points. Devices have various Input mechanisms to command the Controllers.
> Course	The Course OSID provides course management services for educational offerings or other learning units related to an event or curriculum. Courses may be used to model a class, conference tutorial session, or even a meetup group.
> Dictionary	The Dictionary OSID manages key/value pairs. A key and a value may be of an arbitrary type. Dictionaries may be used to support a Locale or Configuration OSID, or may be used to provide any dynamic translation or conversion. The Dictionary OSID is a powerful tool for abstracting and simplifying applications and other OSID implementations.
> Filing	The Filing OSID provides a means for managing and accessing files and directories. The Filing OSID is used to abstract assumptions made about using a specific file system, or can be used to provide a file-based application a file system oriented view of other OSIDs.
> Financials	The Financials OSID provides management of financial data through G/L accounts, activity accounts, budgeting, and posting transactions.
> Forum	The Forum OSID defines threaded discussion groups managing a set of posts and replies.
> Grading	The Grading OSID defines a service to apply grades or ratings.
> Hierarchy	The Hierarchy OSID is an auxiliary service providing a means for accessing and managing hierarchical relationships among OSID Ids.
> Hold	The Hold OSID provides a means for managing blacklists. The Hold OSID adds the concept of “except for” to the OSID environment. The Hold OSID can be used as part of the implementation of an Authorization OSID to override authorization Functions or as part of the Rules OSID to define managed Checks in a business process.
> Id	The Id OSID provides the means for creating and mapping identifiers. All OSID objects are identified by a unique and immutable Id. The Id OSID can be used to generate new Ids when creating new objects
> Inquiry	The Inquiry OSID provides a means for managing a series of inquiries and acknowledgements. An example use of the Inquiry OSID is to capture an acknowledgement from a user before proceeding with some action.

Package	Description
> **Installation**	The Installation OSID manages software packages and installations.
> **Inventory**	The Inventory OSID provides the service of managing inventories of things.
> **Journaling**	The Journaling OSID defines an auxiliary service to manage journals and journal entries for versioning.
> **Learning**	The Learning OSID manages learning objectives. A learning Objective describes measurable learning goals.
> **Locale**	The Locale OSID provides the service of localizing applications defining interfaces for string translation, measurement unit conversion, calendar conversion, spatial conversion, and currency conversion.
> **Logging**	The Logging OSID provides a means to read and write to logs. A Log represents a collection of log entries. Each log entry is composed of a priority Type, timestamp, Agent, the agent's associated Resource, and a record.
> **Mapping**	The Mapping OSID provides a means for managing inventories of places and performing a variety of mapping operations.
> **Messaging**	The Messaging OSID sends and receives messages.
> **Metering**	The Metering OSID is an auxiliary service to read measurements and calculate statistics.
> **Offering**	The Offering OSID defines a means for managing offerings of things, participants in offerings, and their results. This is an OSID expression of the Kuali Student core model.
> **Ontology**	The Ontology OSID is an auxiliary service used to define subject matter that can be related to OsidObjects. Ontologies are an alternative to tagging where structure, restricted vocabulary, or localization of topic names are desired.
> **Orchestration**	The Orchestration OSID serves to bridge multiple OSIDs. A single orchestration service can be instantiated that coordinates the loading of other OSIDs. For example, a repository service may also provide a calendar view of assets. Without the Orchestration OSID, the implementations of both the repository and calendaring providers would need to be specified through the runtime environment. With the Orchestration OSID, a single OSID can be instantiated where the loading of the Repository and Calendaring OSIDs are handled by the orchestration provide.
> **Ordering**	The Ordering OSID provides a means to manage orders and product catalogs.
> **Personnel**	The Personnel OSID defines people and organizations.
> **Process**	The Process OSID is an auxiliary service that manage Processes and mapping of external OSID Ids to States.
> **Profile**	The Profile OSID serves to help applications make decisions based on user attributes. The format and interpretation of the user attributes are abstracted to allow the application to ask questions of the profile service. The Profile OSID can be used to make application decisions regarding localization, accessibility, or roles.

Package	Description
> **Provisioning**	The Provisioning OSID provides a means for requesting and assigning Resources. There are many aspects to this service from the set up of a provisioning system to the acquisition and return of Resources.
> **Proxy**	The Proxy OSID helps a consumer map external data, such as that received via a server request, into a Proxy that can be used with OSID proxy managers. The purpose of this OSID is to modularize assumptions made about the input data into another OSID Provider, such as the authentication or localization information.
> **Recipe**	The Recipe OSID is the how-to service providing a means for managing lists of directions to create or do something. The Recipe OSID can be used in conjunction with the Resourcing OSID and the Tracking OSID to provide instructions or stock answers on accomplishing a variety of tasks. The Recipe OSID may also be used alongside the Learning OSID for managing the skills required to complete various tasks and it may be used in conjunction with the Inventory OSID for managing an inventory of materials required to accomplish jobs. While the Recipe OSID may also be used with the Checklist OSID for a simplified tracking of progress, it can always be used to store your favorite recipes.
> **Recognition**	The Recognition OSID is a service that confers Awards on to Resources. The Recognition OSID may be orchestrated with the Voting OSID for managing nominees.
> **Relationship**	The Relationship OSID provides the ability to relate and manage data between OsidObjects.
> **Repository**	The Repository OSID provides the service of finding and managing digital assets.
> **Resource**	The Resource OSID defines a service to access and manage a directory of objects.
> **Resourcing**	The Resourcing OSID provides a means for resourcing work. The Resourcing OSID works in conjunction with the Workflow OSID, Learning OSID, and Tracking OSID to manage the assignment of Resources to Work as part an an overall work management system.
> **Room**	The Room OSID provides a service for space accounting.
> **Rules**	The Rules OSID provides a means to evaluate rules and retrieve results. A Rule represents a something that can be executed and can be organized into Engines. The rule evaluation accepts a Condition that is used to supply input conditions to the rule engine.
> **Search**	The Search OSID defines an abstract search framework.
> **Sequencing**	The Sequencing OSID is an auxiliary service for sequencing Ids. Sequences may be externally applied to any OsidObject. An OsidObject may participate in multiple sequences, or Chains, that can each be applied and accessed to different users or contexts.
> **Subscription**	The Subscription OSID provides a means for subscribing to communications. Subscribers are represented by Resources. A Subscription service implicitly uses the Contact OSID for accessing Addresses for a Resource subscriber.

Package	Description
> **Topology**	The Topology OSID provides defines a service to manage and connect nodes.
> **Tracking**	The Tracking OSID provides a means for tracking Issues. The Tracking OSID is part of a service cluster that includes the Resourcing OSID, Ontology OSID, Workflow OSID, and Process OSID. Each of these services defines a set of service operations to construct an overall work management system.
> **Transaction**	The Transaction OSID provides the means in which multiple OSID sessions can be coordinated. An OsidSession supports transactions by supporting the Transaction interface. The TransactionSession is the way multiple Transaction objects can be managed.
> **Transport**	The transport OSID provides a means of moving data to or from the local endpoint.
> **Type**	The Type OSID defines a set of interfaces for managing Type definitions. Types are used as an identifier primarily for identification of interface extensions throughout the OSIDs and occasionally used as an extensible enumeration. An agreement between an OSID Consumer and an OSID Provider means they support the same Type.
> **Voting**	The Voting OSID defines a means of holding elections.
> **Workflow**	The Workflow OSID provides a means for managing the flow of work. The Workflow OSID is part of a service cluster that includes the Resourcing OSID, Tracking OSID and Process OSID. The Workflow OSID provides an overall view of the flow of work through a process.

Appendix 3. Flipped Classroom Methodology Detail

Description of Methodology:
A computational aerodynamics class meets twice a week, 1.5 hours per in-class session. Prior to each class, students are expected to read the materials and view short videos covering the topics for the next class session. After reading and viewing, students take a brief conceptual assessment to test their basic understanding of the concepts presented. This homework activity is meant to replace the traditional in-class lecture. There is a date and time, prior to class time, by when all students are expected to have completed the flipped homework assignment.

Prior to class, the professor reviews the results of the conceptual assessment and decides whether to take 10 minutes at the beginning of the in-class session to review particular “fuzzy” topics, or perhaps identify students who may need some extra attention during in-class activity.

During in-class time the students are split into groups of two or three (perhaps organized by shared mis-understandings based on the assessment). Together they work on the problem set for that class session, which they are expected to finish by the end of the class period. The professor and TAs keep an eye on everyone to offer help or to take a mini “lecture-break” to address any common issues they observe.

Overall Service Mappings:
A number of primary service packages will be utilized as part of the EIS suite to support the functionality and data persistence of these applications (a complete set of defined services packages is listed in the appendix):

> *Learning Service Package* - Author, map and manage the learning outcomes for the course. Conceptual assessments, problems in problem sets, and elements of the Syllabus can be related to one or more individual learning outcomes.

> *Repository Service Package* - Authoring, managing, and delivering the learning assets that the students will experience as part of the homework.

> *Assessment Service Package* - Authoring of the conceptual problems, arranging them into assessments, delivering those assessments and tracking and reporting the student responses. Authoring and management of problems and problem sets, to be printed and handed out in class.

> *Course Syllabus Service Package* - Defining the overall structure of the course, authoring the “flipped” homework assignments, grouping them into modules.

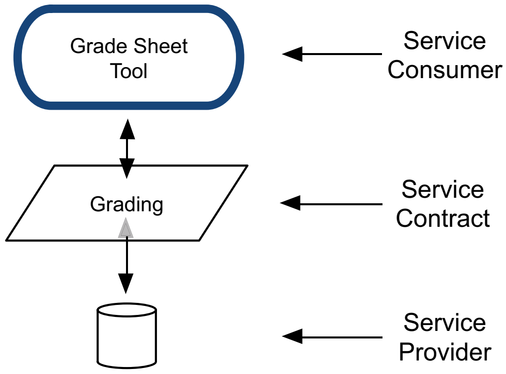
> *Course Plan Service Package* - Managing a particular offering of the syllabus in a particular semester to manage how the structure of the course as defined by the Syllabus service package maps to a particular schedule of the course.

> *Messaging Service Package* - Send and receive messages between users utilizing preferred messaging technologies. In particular for messaging students about information they should know prior to the in-class activity based on flipped homework assignments.

> *Grading Service Package* - Apply and manage grades or ratings.

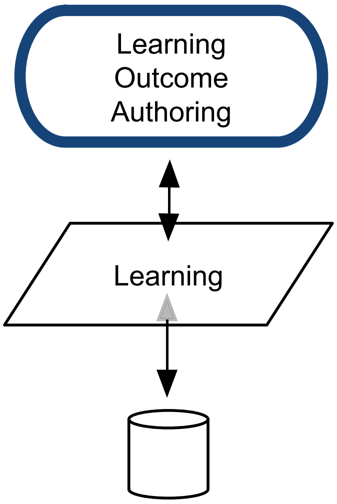
Service Contracts, Providers, and Consumers
Service Contracts, as that term implies, define core functional and business agreements to be entered into between two pieces of software or between developers of such software. A piece of software may fulfil the role of a *Service Provider* or a *Service Consumer,* and more complex software systems, like a virtual learning environment (VLE), may play both roles depending on the circumstances of use. Think of the Service Consumer’s role as that of instigating actions, like “post a grade” or “get a digital asset”, and the Service Provider’s role is to respond to such action as best it can, subject to the permissions of the user, or the limitations of the underlying technology.

For instance, in the graphic below, A Grade Sheet Tool is a Service Consumer of a grading Service Provider, which likely stores grades, columns, gradebooks and other assorted grading related things in a database. Both the Consumer and Provider fulfil their respective roles through the Grading Service Contract:



Key Application Elements for the Simple Flipped Classroom:
Learning Outcome Authoring
The inventory of learning outcomes to be covered through the course needs to be authored, and, optionally,relationships drawn between them. Learning outcomes may be organized hierarchically,

or through requisite/pathway relationships, etc. The Learning service package covers the learning outcome administrative and lookup/query functionality to support this.



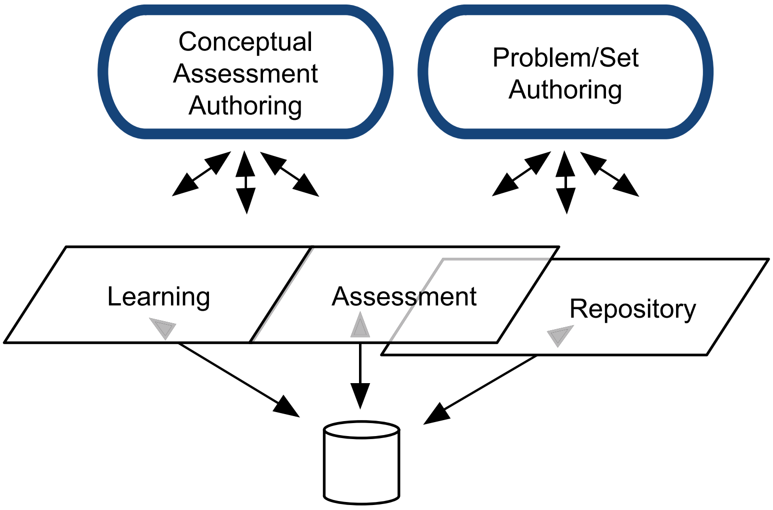
Digging deeper, we can begin to disaggregate an application element even further to identify discrete component functionality that define in detail the user experience. For the Learning Outcome Authoring application component, we might create a component inventory like this:

Component:	Description:
> Hierarchy Traversal	For navigating hierarchies of catalogs or learning outcomes.
> Catalog Admin	For creating, updating and deleting learning outcome banks.
> Learning Objective Admin	For creating, updating and deleting learning outcomes
> Hierarchy Design	For administering parent-child relationships between learning outcome banks or learning outcomes.
> Sequencing	For administering the order of learning outcomes within a hierarchy.
> Requisite Navigation	For administering requisite/depending relationships between learning objectives
> Requisite Assignment	For navigating requisite/depending relationships between learning objectives
> Objective Lookup/Query	For finding and getting learning outcomes

For the remainder of application elements in this example of the Flipped Classroom learning methodology we will not go to this level of detail,

Assessment and Problem/Set Authoring:
The instructor applications for authoring conceptual assessments and problem sets may be exposed through one or more user applications. Three primary service packages will be used to support these applications.

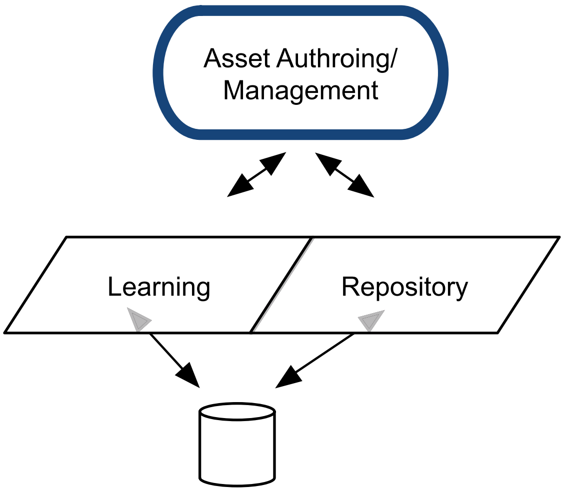
Conceptual Assessments will be designed for online delivery and will likely be automatically evaluated multiple-choice, numerical response, short answer or matching type assessment items. The Assessment service may work hand-in-hand with the Repository service for storing and accessing any images or other assets required for displaying the assessment items.



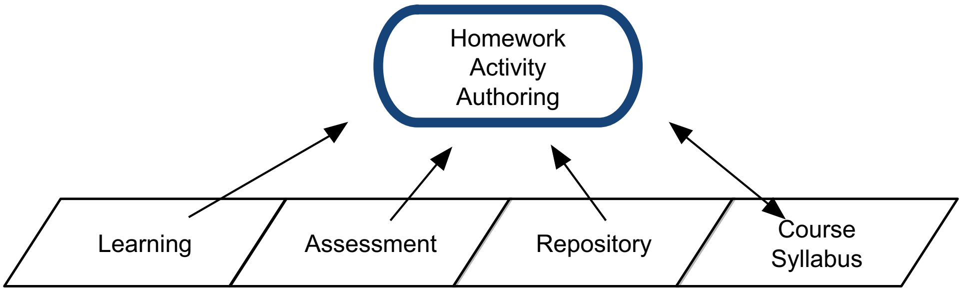
Problems and Problem Sets will also be managed through the assessment service, and since they are intended to be handed out in class may be authored as LaTeX, HTML or other kinds of document formats. Solutions can also be managed through the service.

Both applications would also allow the author to relate conceptual assessments items or problems to learning outcomes managed by the Learning service. This will allow faculty to discover items and problems based on learning outcomes for the purpose of assembling future assessments or problem sets. This can also be used by other pedagogical components which may provide individualized student experiences based on their progress towards particular objectives.

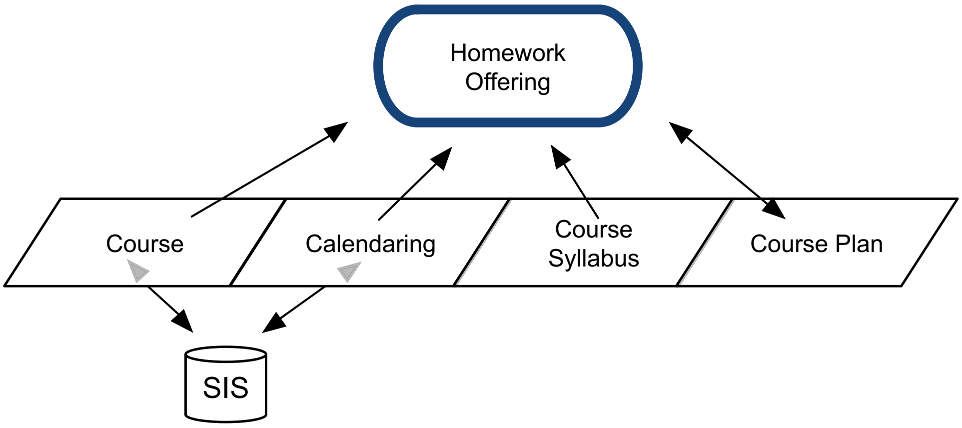
Curricular Asset Authoring and Management:
The flipped homework assignments will be built with curricular materials, including but not limited to readings (PDF, LaTeX, slide presentations, etc), videos, and simulations. These may be authored and uploaded by the instructional staff, or identified as external URLs to existing content. Relationships between these assets and the authored learning outcomes may need to be managed for for more efficient discovery and re-use.



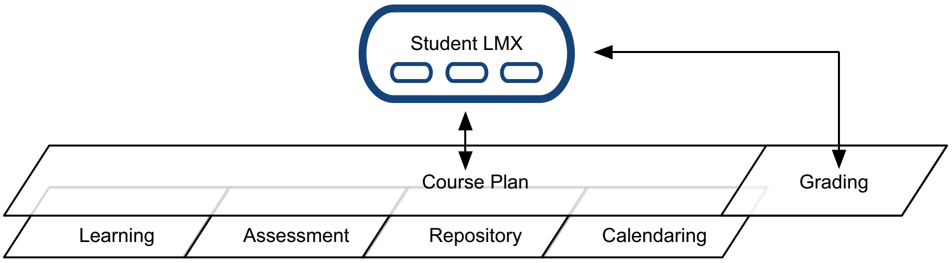
Flipped Homework Activity Authoring Application:
The readings, videos, and associated assessments for a particular "lecture" will need to be organized into a homework activity. The tool for authoring these activities will utilize the Learning service for searching across the set of required learning outcomes and references to Assets in the Repository service and Assessment Items and Assessments in the Assessment service. (Note, we are omitting the service provider side in all the following graphics, as the focus of this exploration is on consumer application elements)



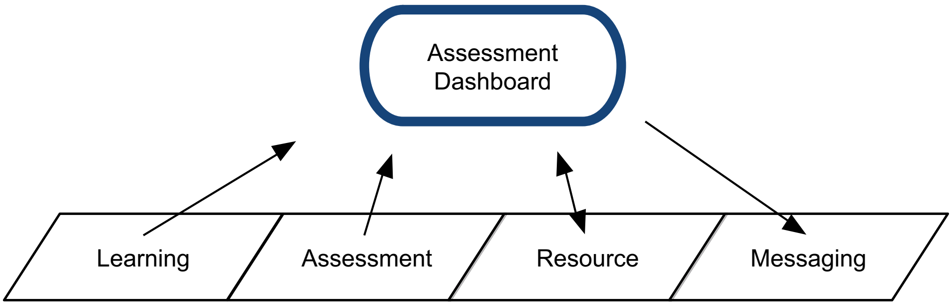
When the author or course instructors decide to offer a run of the course for a particular term (like Fall 2015), a lesson plan will be developed that aligns the homework assignments with the schedule for the course. This is accomplished through the Course Plan service. Information pertaining to the term, and its associated schedules and any holidays, etc, will likely come from a Student Information System (SIS). This may alert the instructor to potential conflicts that need to be resolved, like if it is discovered that there are only 29 in-class sessions for the semester, but the curriculum calls for 30 flipped homework assignments.



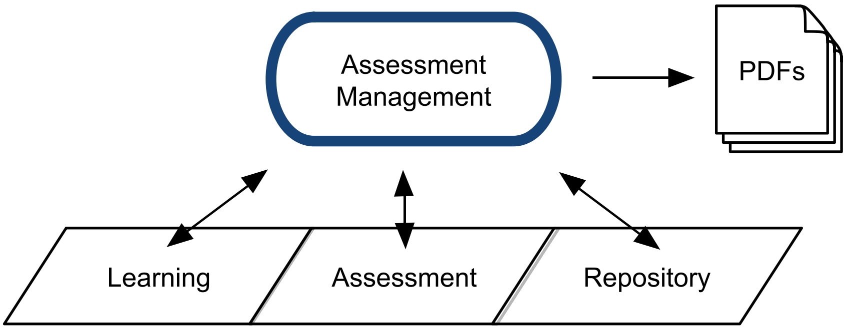
Student Learning Method Experience Tool (LMX):
The Student LMX tool is primarily a consumer of of the Course Plan Service, which, in turn, is drawing from other services where various elements of the activity have been authored and where content is managed. This tool will require various other applications for displaying content and delivering assessments and providing the student with a way to display information about the learning outcomes being addressed. Students will also be automatically graded on whether they did or did not attempt all the questions, which requires the LMX creating grade entries..



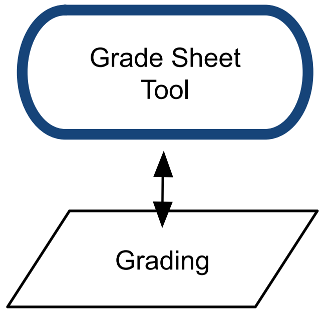
Instructor Assessment Dashboard (review students' conceptual grasp):
The Instructor Assessment Dashboard allows the instructional staff to review all the students' responses to conceptual questions attempted during the flipped homework activity. "Fuzzy" concepts can be identified and flagged for review in class. Optionally, students can be organized into problem set groups based on similar or differing conceptual blocks, and messages will be delivered via each student's preferred method to indicate who they will be matched with for the in-class work.



Assessment Management Tool (assemble problem sets):
Instructional staff will use an Assessment Management application to assemble a problem set or problem sets for delivery in class. These will be exported as PDF for printing out.
The Assessment Management Tool may very well be the same overall user interface as the assessment authoring tools covered earlier



Grade Sheet Tool:
Finally, following the in-class problem set activity, instructional staff will collect the problem sets from each group and enter grades into the gradebook. This tool also allows the staff to report on which students are consistently completing the flipped homework assignments, which has been automatically entered.



uoc.edu @UOCestudiant @UOCestudiante @UOCuniversity youtube.com/UOC facebook.com/UOC.universitat

Headquarters
Avinguda del Tibidabo, 39-43
08035 Barcelona
Tel.: 93 253 23 00

All institutional offices at **offices.uoc.edu**

Universitat Oberta de Catalunya