Abstract. We describe the OpenACS web application framework and dotLRN, its
Learning and Community Management System. dotLRN is an ongoing open source
project to deliver a complete course and community management infrastructure that
is scalable, robust, extendable and complies to open standards such as SCORM. The
software, which is currently used to deliver online courses and collaboration tools at
a number of major universities, is based on the OpenACS platform that implements
an object oriented data model on a relational database. dotLRN includes
functionalities for the school portal, student and class management, calendars,
forums, file sharing, etc. In this paper we discuss the architecture, the history, and
the future of the OpenACS/dotLRN system. We also provide five case studies of
institutions adopting it.

Keywords: Learning Management System, Intelligent Tutoring Systems, Software
Engineering, Object Oriented Architectures

1. Introduction

With the increase of student enrolments, pressure in universities’ budgets and the
potential of new technologies for education, e-learning is becoming ever more prevalent. It
is believed that learning and community management systems, that underlie e-learning
initiatives, are to become part of the core for all educational institutions. This perception
of the future has made many universities and large organizations aware of the risk of vendor
lock-ins and maintaining the value of their most important assets - their training processes
and materials. Institutions do not want these assets to be pegged to a single company. Two
complementary approaches have been taken by many of these institutions: open source
software and open standards.

Open source software is increasingly popular in institutions that have a tradition of
sharing knowledge, as is the case for schools, universities, and other publicly funded
institutions. Moreover, many organizations often like to tailor their information systems
to their own business processes and they benefit from open source software, since it allows
them to customize it thoroughly. Open source licensing is used to distribute software
implementations in a very flexible way: it allows organizations to copy, modify and
distribute it without a licensing cost, normally, with the only condition that the software be
distributed to others under the same licensing conditions.

Open source allows the user institutions to reuse and customize the design and
implementation done at other institutions, and even build into it novel features, as if they
owned the system. Open source development has been adopted by many as a way of
allowing customisability and reusability, and at the same time reduce licensing costs. Open
source has big advantages over purely home grown systems. Building in house is often
regarded as expensive and risky, because it doesn’t take advantage of the economies of
scale that make software cheaper. Comparing to build the system from scratch and in-
house, open source provides the advantages of proprietary software: lower (compared to
both) initial investment and low ongoing development costs, but it has the high level of
ownership and independence of in-house systems.

In addition to open source, a number of open standards are being developed so
organizations can maintain a certain level of independence and interoperability. As an
example of these open standards, we can emphasize the Open Knowledge Initiative (OKI),
that aims to define an open architecture specification to support the development of
educational software [VIII]; uPortal, a framework for producing a campus wide portal [IX];
IMS Open Specification Standards from IMS Global Learning Consortium [II]; and The
Sharable Content Object Reference Model (SCORM).

The above two approaches: open standards and open source software are becoming
increasingly important and are being used to build the e-learning infrastructure of
universities around the world. E-learning is increasingly connected to the collaboration
tools that academics use, therefore building tools that integrate both seamlessly is a
worthwhile effort.

In this article we describe the OpenACS web application framework and dotLRN
[V], an open source e-learning platform currently deployed to support online courses and
academic research by several important institutions, including the Sloan School of
Management at the Massachusetts Institute of Technology (MIT, USA), Heidelberg
University (Germany), the Cambridge University (UK) and Universidad Nacional de
Educación a Distancia (UNED, Spain). In dotLRN, a number of collaborative applications
are seamlessly integrated using a single data model. The system is implemented on the
OpenACS web application framework [2], [IV], also described here.

In section 2 we will discuss the system design and architecture of OpenACS and
dotLRN, in section 3 we will present some case studies of institutions using the system and
in section 4 we conclude.

2. dotLRN: System Design and Architecture

dotLRN was originally built by ArsDigita corporation (now part of RedHat) for the
Sloan School of Management at MIT. The system was based on the ArsDigita Community
System (ACS), which is a toolkit used by many large organizations such as Siemens, the
World Bank and Levis. The original learning management system was called ACES [6],
[9], [1] and distributed under the GPL open source license. The second version was
released two years later under the name of dotLRN, with a new architecture described here.

2.1 System Design

The initial ACS platform was implemented on the AOLserver web server using the
Tcl programming language and the Oracle RDBMS. In December 1999, the OpenACS
project was commenced by a number of developers seeking to implement a version of ACS
that would run on an open source RDBMS. Today, OpenACS is a mature project in its own
right, with over 6000 members and at least 10 companies that provide commercial support
and development.

As discussed, dotLRN is built using the OpenACS object oriented web application
framework [2],[1],[IV] and this is designed to improve the reusability of design and
implementation following software engineering practices [3],[4],[5]. OpenACS provides a
standard data model and API that are briefly described in the next section. The framework’s
object oriented architecture includes a content repository (used by most applications that
need to store “content”), a user/group management module used by a permission system, a
request processor that enables persistence, authentication and templates.

In OpenACS, when a page is requested by a user, the request processor
authenticates the user, then utilizes the user/group module (part of the core system) to
authorize or not the request. If authorized, the page is custom made using the template
system and the user model. The template system assembles the page from a number of
content snippets, and this is sent by the server to the user. The template system assembles
2.2 dotLRN Architecture

dotLRN features a number of collaborative applications such as calendars, forums, file sharing, assessment tools, etc. that add up to the ones that the OpenACS framework provides (i.e. e-commerce, content management, workflow, webmail, etc). Many of these functionalities are similar to the ones found in other LMS such as WebCT and Blackboard, sometimes with additional features made possible by the common data model (e.g. a centralized alert system), sometimes with less due to the shorter product history. For the sake of brevity we will not discuss them here.

The OpenACS v4.6 framework architecture is described in Figure 1. The diagram has five main components: user, webserver, framework, database and services. The dotLRN architecture is similar in many ways to WebCT Vista [VII], but it uses OpenACS instead of BEA’s Weblogic framework, and it may use other databases besides Oracle, particularly PostgreSQL.

2.2.1 dotLRN user portal

The OpenACS architecture shown in Figure 1 is composed of the acs-core packages which provide an API for common functionality among other services and packages. The services are packages that do not interact with the user and establish an interface between the packages, where the actual application resides. Since dotLRN is built on the OpenACS framework, it inherits all OpenACS functionalities, modularity and features.

As an example, dotLRN uses the OpenACS calendar. Several calendar instances can be bundled in an individual student’s portal, displaying in a single page all the events for each course the student is enrolled in. Each view of a package (or piece of content) is called a “portlet”. The system adapts the portal to each user by gathering information about the courses and communities that the user is enrolled or belongs to, and which applications these courses and communities are using. Each of these applications will then generate appropriate content shown in a portlet. It can also be personalized by the user who can add and remove these and other portlets.
2.2.2 dotLRN communities

In a physical world institution, learners learn through the courses they take and the communities they interact in. In a company, the communities could represent special interest groups, or projects or just social gatherings. In the university, clubs and societies are very common. Both types of activities are supported in dotLRN, so all dotLRN applications may be used in either. This requires very clear scoping of each application so each course/community can have its own set applications, templates and permissions. dotLRN uses a “subsites” concept that separates contexts between the different types of communities (“classes” and “clubs”). Each subsite instance (“dotLRN_community”) may have its own permission structure (i.e. different people in different roles). Figure 2 shows in a class diagram how dotLRN communities can be classes or clubs, both types providing different sets of collaboration tools, roles for people and default templates. In both cases, these communities have their own instantiation of calendars, forums, news, groups, file sharing, etc. Figure 2 also shows this structure and how classes relate to the department and subject classes. “departments” describes the standard organizational structure within the institution, “subject” contains information that repeats over different instances of a “class”. “classes” specify people and their roles, start and finish dates, and other attributes that are specific to that instance. For example, “Web Application Frameworks” (a subject) could be taught at the “Department of Computer Science” (a department) in the second semester of 2002, by Juan Teacher, under the name “Web Application Frameworks” (a class).
2.2.3 Portlets

dotLRN uses the OpenACS Portal architecture to generate the user’s portal interface. dotLRN portlets implement the appropriate interface between OpenACS applications and the dotLRN portal. By doing so, these portlets deliver a customized functionality based on OpenACS applications. Figure 3 shows a class diagram for the portlet’s design.

2.2.4 dotLRN users

OpenACS can support complex user hierarchies by defining “parties” that may be: “users” identifiable by a distinct email, or a “person” with distinctive first and last name, or a “group” that maybe one or more of the above, or even other groups. These relationships are shown in Figure 4.

With this architecture dotLRN users can hold more than one particular role. For example, a user could be a teaching assistant in one class and a student in another. Therefore, the role of a user in a class would determine what the user is allowed to do in it. All the names of the roles and entities in dotLRN, such as class, communities, instructor, etc, can be easily changed to the needs of the institution or country where the system is used. The dotLRN class is a subclass of the OpenACS parties with “role” as an attribute that can take values such as Student, Course Assistant, Instructor, etc.
Face to face or mixed mode courses normally vary in length, structure and contents supporting predefined learning outcomes. Purely online courses may go further and change for each student. Using student and learning models each student may be experience an adaptation of the original course. These features are called adaptive learning and are one of our research goals with dotLRN.

![Figure 4: OpenACS Parties package](image)

### 2.2.5 OpenACS Content Repository and workflow

Web applications manage large amounts of little pieces of content. These pieces include postings in discussion forums, events in a calendar, news, attachments and even multimedia elements. In OpenACS every piece of content is registered in a content repository that standardizes the way applications access and manage this information. This additional layer of complexity pays off by making possible the reusability of all content management functionalities such as permissions, formatting or other novel functionalities that may have not been included at the time of designing an application. A service created for the content repository, such as a information retrieval or automatic document classification system, can automatically be used by all applications in the system. With a different architecture, the service would need to be re-implemented or adapted for each single application, making it aware of its internal representation.

The OpenACS workflow service also adds useful capabilities to dotLRN since it can be used to make applications “workflow aware” in a standardized way. The workflow enabled application is able to use all the alert and event management capabilities built into the workflow package. The workflow system could arrange emails or SMS alerts to be sent when a task has been finished or needs to be started, it could arrange for the tasks to show in the users’ centralized calendar or it could provide applications with features that were not considered at the time of designing them.

### 2.2.6 OpenACS Internationalization

Through a collaboration between Sloan School of Management, Heidelberg University and Greenpeace International, OpenACS and dotLRN are being translated and internationalised to a number of languages (locales). The internationalization project will allow dotLRN to be available in a number of languages including English, Spanish, and German. It will have enhanced support for running courses over different timezones. In order to achieve this a number of issues are being addressed including: scalability, character sets, translations and message catalogues.
2.2.7 dotLRN and SCORM

As discussed in the introduction, open standards compliance is an important requirement for e-learning infrastructure. The OpenACS community is committed to open standards, most recently a lot of effort is going into making dotLRN complaint with SCORM 1.3 standards. This should be available in July 2003.

The Sharable Content Reference Model (SCORM) is one of the most important of these initiatives and it aims to provide:

- **Accessibility**, the ability to access training from remote locations and deliver trainings in any location.
- **Interoperability**, a training or course developed in any platform can be taken in any other platform.
- **Durability**, despite changes on the course delivery technologies, the course should be delivered properly as it was conceived.
- **Reusability**, trainings or courses can be reused in other contexts and/or courses.

The SCORM is a collection of specifications from industry groups such as the IMS Global Learning Consortium, Inc., the Aviation Industry CBT (Computer Based Training) Committee (AICC), the Alliance of Remote Instructional Authoring & Distribution Networks from Europe (ARIADNE) and the Institute of Electrical and Electronics Engineers (IEEE) Learning Technologies Standards Committee (LTSC).

SCORM defines a conceptual model for Web-based learning based on its “Content Aggregation Model” and “Runtime Environment.”

![SCORM Diagram](image)

The implementation of SCORM 1.3 in dotLRN, is being developed by Sussdorff-Roy GbR (Germany) and the Web Engineering Group (University of Sydney). The project includes the following functionalities:

- **Import/export of SCORM compliant learning objects**
  SCORM compliant courses can be seamlessly imported into dotLRN and be deployed to a particular class or community maintaining dotLRN native permissioning system. The addition of IMS Content Packaging specifications, allows dotLRN to export courses and classes following the standard so it can be use in other e-learning platforms that are IMS Content Packaging and/or SCORM complaint.

- **A tool to manage SCORM courses within the OpenACS/dotLRN framework**, An administrative tool to manage SCORM compliant courses, including a full metadata search of all SCORM courses (content aggregations), sequences
(activities), SCO/As (learning objects), and assets (files). This tool facilitates finding learning resources to be reused by other courses (content aggregations). Regrouping these learning resources in different courses that serve different purposes fulfils the final goal of SCORM. Given the flexibility of dotLRN content repository, SCORM learning objects are shared among lectures who want to create their own content aggregation or courses. Moreover, and leveraging on dotLRN robust architecture, lecturers and/or content developers can take advantage of automatic versioning of their learning resources.

- A runtime environment (launching and delivering SCOs and tracking)
  In order to deliver web content learning objects and track the results achieved by the learners, dotLRN is fully compliant with SCORM runtime environment to launch and deploy learning resources on a web browser. Leveraging on the OpenACS/dotLRN internationalization functionality, SCORM delivers would deliver customize course content in the learner’s default locale.

- A learning objects sequencing engine
  SCORM 1.3 includes a complex sequencing engine for delivering SCORM learning objects based on IMS Simple Sequencing Specifications (do we need to reference this?). This sophisticated sequencing engine allows lectures and/or content developer to create complex sequencing structures of learning objects that accommodate to the learner’s learning performance.

3. Case studies

3.1 Sloan School of Management/ Massachusetts Institute of Technology

Sloan is MIT’s business school and the main driver for the dotLRN project. The original developer, ArsDigita Corp. collaborated with the Sloan School of Management writing the original implementation of dotLRN, formerly called the Arsdigita Community and Educational Solution (ACES). Eight classes used the system as part of a pilot program in Fall 2000, leading to a user community of approximately 700 students and faculty. After the successful pilot, Sloan extended its use to all its courses through SloanSpace v1.

MIT’s is a world leader in many areas, particularly in sharing software and ideas. DotLRN, together with the Open Knowledge Initiative, the Dspace digital library project and the Open Courseware project are part of its Council of Educational Technology (http://web.mit.edu/cet/), all making their outcomes available to the community or the so-called “Intellectual Commons”. Since the origins of dotLRN, Sloan has maintained the software available to everyone under the GPL license. Even though several universities are already using it, the first official release of dotLRN 1.0 will happen in July 2003. dotLRN 1.0 is the software behind SloanSpace v2.0.

Sloan is also coordinating a management structure around the project with a technical advisory board that provides guidelines on architectural and implementation issues, a user advisory board provides recommendations from the user perspective and Executive Board that mostly looks after funding and provides Universities with means for solidifying the ownership of the project.

SloanSpace’s dotLRN infrastructure runs on Solaris servers and Oracle.

3.2 Heidelberg University

The University of Heidelberg is over 600 years old, the oldest university in Germany and one of the oldest in Europe. It has almost 30,000 students every year and its academics have included 8 Nobel Prize winners.
The first standard e-learning platform in Heidelberg was WebCT. The university started tests in 2000 and went live with a number of courses in 2001. Soon after its launch, the academics and administration of the university believed that the software was designed around a North American educational paradigm and that WebCT’s adaptation to the European market had little to do with the examination of different university traditions, but was limited to a one-to-one translation of the software. This became even more obvious with the sudden change in licensing, which would have resulted in a many fold increase in costs for the same functionality. Since in Germany education is paid by the state it is not possible to offset long-term expenses by slight increases in tuition or fees at one university. It suddenly became difficult to justify spending such a large amount of money without any influence over future development. The licensing restricted us from making modifications to the software to better fit our needs.

Carl Robert Blesius who contributed all the information for this case study explains that the university was looking for an investment in infrastructure that would also benefit the whole.

Since sharing new ways of teaching and learning across institutions and national borders is nothing new for universities, they considered that this shouldn’t be any different. The university did not want this tradition of collaboration and sharing to be limited by licensing terms: “Collaboration in learning and research is essential for us and we want to retain some control over the tools we use for this. The tools should be subject to peer review and shared (which again is not a new concept for universities), they should not be trade secrets.”

In the first stage dotLRN will serve 2-3,000 users that are being migrated this year. Soon after it is expected that all 25,000-35,000 faculty, students and staff will have access to the system. This year they are running 100-150 courses.

Heidelberg is the main driver behind the internationalisation effort (i18n) and is improving the support for External Authentication, extending the Wimpy Point presentations package, improving group registration options, building a Curriculum package and Workflow integration.

Heidelberg’s production system were planned to be running dotLRN on Solaris and Oracle.

Due to the success of the tests on Linux and Postgres they expect to use a combination of Solaris and Linux. Reducing costs but maintaining the highest standards for the most critical applications.

3.3 University of Cambridge

The University of Cambridge in the United Kingdom is one of the oldest Universities in the world. It has a world-wide reputation for outstanding academic achievement and the high quality of research undertaken in a wide range of science and arts subjects. The University pioneers work in the understanding of disease, the creation of new materials, advances in telecommunications and research into the origins of the universe. It trains doctors, vets, architects, engineers and teachers. The University's achievements in the sciences can be measured by the sixty or more Nobel Prizes awarded to its members over the years.

Over the past 15 years, the University of Cambridge has been developing online resources to support teaching. More recently, the University has been involved several initiatives to develop standards such as IMS and OKI (Open Knowledge Initiative).

The University of Cambridge in the UK started in 2003 running a pilot programme to assess the feasibility of using dotLRN as their main e-learning and collaboration platform.
Although Cambridge is currently not using dotLRN to support teaching at the moment, it has deployed it to support research and other campus communities with good possibility for eventually become the basis of their course and content management. Today, dotLRN at Cambridge has 300 users and it is expected to grow quite rapidly.

In addition, Cambridge has acquired Coursework from Stanford, Stellar from MIT and it has few in-house solutions. In order to find a system that best suits their needs, Cambridge has deployed Blackboard Basic Edition and also experimented with JetSpeed (Open Source implementation of an Enterprise Information Portal) and followed the development of University of Michigan’s Comprehensive Collaborative Framework (CHEF).

Cambridge has several reasons to work with Open Source initiatives: primary, the ability to modify and extend the software to tailor it to their educational processes. John Norman, director of Cambridge’s Clinical and Biomedical Computing Unit (CBCU), emphasizes that “… a successful open source process actually results in more robust and useable software (although how to get a good process going is not trivial of given).”

### 3.4 Universidad Nacional de Educación a Distancia

Universidad Nacional de Educación a Distancia (UNED), based in Spain, is the most important distance education university in the Spanish speaking world. Since 2000, when they ran their first pilot programme, their main e-learning platform has been WebCT.

Innova Desarrollo Tecnológico (http://www.innova.uned.es/index.adp) is a unit set up by the University’s Institute of Distance Education (IUED) to develop innovative technology as is needed for quality distance education. The Institute teaches continuing education, postgraduate and adult education programmes within UNED. Innova chose dotLRN as their main platform for the following reasons:
- Flexibility, since it could be adapted to the language and university’s processes;
- Ease of administration in contrast to WebCT which they found difficult to administer.
- Cost advantage, as compared to the recently increased licensing costs of the other available platforms.

They are now using dotLRN to teach to 4,500 students in 16 courses through 500 sessions a day. The system runs on a Linux and Oracle platform.

Innova is also collaborating in the internationalisation efforts with the OpenACS community. The dotLRN system being used now is based on ACES and has been translated to Spanish. Innova is working in a making a Spanish version of the new dotLRN also available in the short term. Innova is also working on a document management system using XML-DocBook and EML.

### 3.5 University of Sydney – Web Engineering Group

The Web Engineering Group at the University of Sydney teaches undergraduate courses in e-commerce and software engineering. The group has developed several packages for OpenACS. It is the common platform for research projects in e-learning, knowledge management and language technologies. DotLRN is also used as knowledge management and collaboration tool by the members of the group and collaborators.

As a research project to compare student perceptions on different e-learning systems, we will use the “Introduction to e-commerce Systems” unit [ref], a course in the first semester of 2003, as a sample. The course has an enrolment of about 220 students, and it will be divided in two groups, one using the University supported platform (WebCT) and the other using dotLRN. At the end of the course a survey will be used to measure students’ perception of each platform, and how it supported their learning.
4. Conclusion

We have described the OpenACS web application framework and the dotLRN e-learning system, their basic features and architecture. The system is licensed under the GPL open source license and is being used at several universities around the world. We described the content repository and workflow system were we are planning future work integrating machine learning models used in managing learning objects and processes.

dotLRN is built using the OpenACS web application framework, and designed for reusability of design and implementation. We have showed how the architecture reduces the effort of building new functionalities. In fact, the architecture is similar to other commercial products, but may use a fully open source application stack, including database and application framework.

Acknowledgements
The authors want to acknowledge Carl Robert Blessius from Heidelberg, John Norman from Cambridge and Emmanuelle Raffenne from UNED, and Caroline Meeks for their help in the writing of the case studies.

References

World Wide Web References
[IV] OpenACS Project and documentation [http://www.openacs.org]
[V] dotLRN Project and documentation [http://www.dotlrn.org]
[VI] openfts Project and documentation [http://openfts.sourceforge.net]
[IX] uPortal Project and description [http://mis105.mis.udel.edu/jasig/uportal/faq.html#1_1]