UCLA Sakai Pilot Report to the FCET

Prepared by Project Co-Chairs
S. Kumar, The Anderson School
Rose Rocchio, The Office of Information Technology
June 9th, 2006

The UCLA Sakai Pilot involved 37 courses and was supported by 9 part-time staff members from across four different departments. Based on these experiences, the Pilot team gained experience to be able to answer the questions posed in the FCET's March 2, 2005 Request for Proposal for departmental participation in a Sakai pilot. (See Appendix A for copy of RFP.)

I. Pilot Objectives:

The UCLA Sakai pilot project was authorized in March 2004 to evaluate the new collaborative learning environment (CLE) that was at that time being jointly developed by Michigan, Indiana, Stanford, and MIT funded by a Carnegie Melon grant. The UCLA pilot had four objectives:

- Install and evaluate each new release of Sakai for courses and groups willing to participate/pilot the system
- Integrate the Sakai system with the campus infrastructure (ISIS, The Registrar, etc.)
- Evaluate the viability of the Sakai community
- Provide answers to the questions initially asked of the team by the FCET.

1.1 Participants

- The Anderson School, Center for Digital Humanities, and Academic Technology Services were the initial core pilot team responding to the RFP. (See Appendix B for copy of Proposal.)
- Office of Instructional Development and several individuals joined the pilot as soon as it was initiated. (See Appendix C for complete list of participants.)
- Sakai has been used for courses and in support of project workgroups, and the team has been involved in broad spectrum of activities. (See Appendix D for Sakai National and Pilot Project Milestones.)

1.2 Process

- All of the departments worked together, meeting weekly to discuss technical and support issues and collaborate on solutions.
- Only the primary roles of people are listed.
- Everyone participated in keeping up with advances in the software, and figuring out solutions to problems.
- Many of the team members participated in the National Sakai Workgroup efforts (The Requirements workgroup, the QA workgroup, the UI workgroup).

- Many of the participants went to the various Sakai conferences over the last two years.

1.3 Limited Resources

- While the team has had approximately 12 participants all work was done on a volunteer and "as needed" basis – with no additional compensation to the team members. With the exception of the System Administrator (Pete Nielsen) most participants were dedicated to the project at the level of less than 10% of their time.
- Total resources of the project did not exceed 2 FTE

1.4 Challenges:

- The pilot project has been a challenge as we have decided to implement the 3 major releases of the Sakai software in the last 15 months.
- This has entailed re-integrating Sakai with the campus infrastructure 3 times as our integration code has broken with each new release.
- The next major release of Sakai 2.2 on June 15th, 2006 will again most likely require re-work of our integration code.

II. The Sakai Pilot's Answers to the FCET's questions

Answers by the Sakai Pilot Project Team
(Rose Rocchio, S. Kumar, Jim Williamson, Jim Watkin, Pete Nielsen, Howard Kim, Mike Franks, Annelie Chapman and Harsh Desai)

1. What amount of JAVA and other expertise is needed for a unit to install and run the framework & core functions?

1. Java expertise will be required in a production environment where clustering servers, setting parameters, and monitoring performance are necessary.
2. Additional expertise needed to install and run Sakai in production includes:
    a. Ability to install and configure the server with the Linux OS (Unix experience)
    b. Ability to set up, performance tune and monitor Apache & the mail server James
    c. Ability to set up, performance tune and monitor Tomcat
    d. DBA expertise and the ability to setup and configure MySql or ORACLE
3. Java expertise (5+ yrs & J2EE) is also needed for two additional activities:
    a. Integrating Sakai with ISIS
    b. Integrating Sakai with the Registrar data for courses and rosters

    Our experience to-date is that this requires re-integration with every new version of Sakai as the platform and inter-connectivity capabilities are still being updated.

2. What amount of JAVA and other expertise is needed for a unit to develop and integrate instructional tools?

1. Currently, to develop and integrate tools a developer must have a high competence in Java (see Java expertise description above)
2. With a minimal integration level, other tools built with PHP, Python etc. can be used within Sakai by using iframes, and possibly the tool interoperability specification
3. To achieve a high-level of integration with Sakai and its "bundled" tools, cross-language interoperability is more of a goal than a reality at this stage. It is however, a strategic direction of Sakai to allow for this type of interoperability.
4. IMS has created guidelines for Tools interoperability which are available here: http://www.imsglobal.org/ti/tiv1p0pd/imsti_guidev1p0pd.html
5. Additionally, there is a Sakai effort to allow tools from different Learning Management Systems (Blackboard, WebCT, Moodle & Sakai) to interoperate using the IMS guidelines. To read more about this effort see this link (see the presentation at): http://bugs.sakaiproject.org/confluence/pages/viewpage.action?pageId=9668
6. The IMS TIF (Tool Interoperability Framework) Launcher Tool will be available in Sakai 2.2 (due out this summer?)

3. What does testing tell us about the robustness of the underlying architecture (speed, capacity, security, reliability, etc.) for the UCLA environment?

1. In pilot mode, running Linux (the OS), Apache (the http server), Tomcat (the web server), MySQL (the database) and Sakai (the application layer) all on one server has had relatively few problems. We have however, encountered a few nagging bugs (UTF-8).
2. With the proper architecture in place (8 web servers + 1 database server running Oracle), several schools including Indiana, U Michigan and Berkeley have rolled Sakai out to large campus-wide populations.
3. However, we have learned from the community, that considerable effort has gone into "performance tuning" campus installations of Sakai. Especially where the integration with each campus's Registrar's system and authentication/identity management are concerned.
4. Additionally, we are aware from Rutgers experience, that performance tuning of MySql is needed (to the level of having to rewrite database queries) to get Sakai to perform at an acceptable level for a large number of users. Most of the initial Core (4 schools involved in Sakai Project) installations of Sakai are using Oracle, thus original queries were written to take advantage of Oracle's features therefore additional database tuning does not seem to be required.

4. What is required to make SAKAI interoperate with UCLA's core systems and processes (minimally ISIS, online gradebook, course rosters, instructional materials repositories, library, other?)

1. Currently, we have integrated Sakai with ISIS and the Registrar's course rosters. However, Sakai is still very much in flux and the result of this is that the API's are not stable. Thus with each new version of Sakai that has been released, we have had to re-develop our UCLA integration code. Intimate knowledge of Sakai and an ongoing relationship with the community is needed to keep this code working as new versions are released.

2. A skilled Java developer is required to figure out a way to code a java subclass that replaces the core Sakai class to do what we want it to (like accept a login from ISIS)

3. To interoperate with the My.Ucla gradebook there are varying levels of interoperability that could be achieved:

Level of Integration	**Description**	**Status of integration**
Minimal Level or "Loose integration"	Include as a web content tool (within an iframe) the My.Ucla gradebook or gradebook Express	Achieved. No login needed as both Sakai and My.Ucla use ISIS. But the webpage does not link directly into a specific class.
Moderate Level or "partial integration"	Allow professor to click a button that would behind the scenes send grades to the My.Ucla backend – which would then submit them to the Registrar	Not achieved or attempted with the pilot project. The developers of the Sakai gradebook (from Berkeley) advised us that the gradebook was changing substantially with the new release of 2.1 and warned us that much of our code would need to be re-written if we attempted
Highest Level or "tight integration"	Allow a professor to record all grades of quizzes or tests or assignments in Sakai and have them replicated in the My.Ucla detailed gradebook. A high level of	Not achieved or attempted with the pilot project. The My.Ucla Gradebook developers would need to cooperate with UCLA Sakai developers (TBD) to enable these two very different environments to work together. In the absence of this success,

	integration would surely require that grades could be passed back from the registrar to Sakai as well (eg. for adaptive learning or student portfolio, or simply a professor prefers to use the MyUCLA gradebook but wants the results viewable in Sakai)	the Registrar would need to allow the Sakai Gradebook to directly add grades to the Registrar system. This will entail formal university approval and process, then cooperative work between the Registrar's developers and Sakai developers. While the experiences at other schools is good to know, we would hesitate from making an implicit suggestion that the ability to X, Y, or Z at another university means UCLA would be able to do so easily. The differences in policies and technical environments will dictate the potential applicability of others' experiences.

4. Regarding integration with instructional materials repositories:

1. Sakai's Twin Peaks tool has illustrated Sakai's capability with simple (i.e., no security or subscription exchange) integration with three data sources. For example, as Giunti Labs Lobster has an OKI DR plug-in for Lobster, Sakai can query that repository.
2. Like the Gradebook, integrating UCLA Library resources will also be determined by policies, commercial subscriptions, and technical development and has not been explored.
3. MIT has hired a programmer to integrate Twin Peaks with Dspace, but the status of this work is not well known at this time.
4. We understand that Berkeley is developing a new tool for Sakai to be able to be integrated with instructional materials repositories (images)
5. The library – we understand that Indiana received a 450k grant to look into expanding library integration beyond "public" repositories.

5. What is required to integrate web services into the SAKAI architecture?

Some background on simple web services:

Web services work as a communication between two systems, a "provider" (which provides data) and a "consumer" (which uses the data). A fictional example would be a Testing Tool in Sakai (provider) which would post grades (data) into the My.Ucla Gradebook (the consumer). (This must be a fictional example for the time being because neither the Sakai nor myUCLA gradebooks are able to acts as providers.) The details about which system initiates the communication and how the data is transferred must be worked out by the policies and technical capabilities of each system. For example, does the Testing Tool send the grades, or does the myUCLA Gradebook ask for the grades? How is security handled between the systems? Which security protocols or data formats are supported by each system?

Although web services seems to be a simple idea, in order to integrate web services, very specific exchanges of data must be worked out in an atmosphere of agreed policies, procedures, and technical development.

1. With that background, while limited web service capability is operable within Sakai on a test basis (which further development should expand), the specific web service functions need to be defined in Sakai and UCLA developers would need to write the code to get or send data.
2. The following is from the Sakai 2.1.1004 documentation:

3. Regarding WSRP (Web Services for Remote Portlets) Sakai is a producer but not currently a consumer, i.e., a Sakai tool can be taken into another portal but an existing (non-Sakai) portlet cannot be brought into Sakai.

6. What is required to integrate tools that are currently being used by faculty in various academic units?

1. See the answers to question #4 – integration levels vary from loosely integrated to tightly integrated.
2. In a very simple fashion, tools hosted elsewhere (not on the Sakai server), can be linked into Sakai using a simple "iframe" or portlet window. This does not however imply that the tool is integrated beyond "simple navigational appearances" with Sakai.
3. However, Sakai is a framework and current tools being used by faculty, such as a WebCT tool (WebCT has announced that they were providing the Tool Interoperability Runtime (which works with Sakai) as a "powerlink") that may potentially work within Sakai in the future.
4. From a license perspective, code developed under GPL (Gnu Public License) cannot be further enhanced within Sakai without being "wrapped" as a black-box.
5. Sakai does integrate with a tool that many professors use called "WebDAV" as a way to move around teaching and learning resources. However, WebDAV and ISIS do not integrate at all so we are limited to using WebDAV with one of two workaround solutions we have found for use with Sakai. However, even with this issue, WebDAV in Sakai is fraught with bugs at the moment.
6. As for WSRP (Web Services for Remote Portlets), see the response to question #5; each side of the system must be prepared to offer these services. The status of WSRP in Sakai today is unclear, what is clear is that it is not at the top of the priority list for the time being.

7. What is required to customize the look and feel to accommodate a diversity of unit and personal specifications?

1. Customization as a concept can happen on various levels: look, feel and navigation, icons and tools.
2. Sakai is configurable on a system-wide basis, and requires experienced designers and developers to do so. For example, because of some global settings, individual group designs may conflict within a single Sakai system. Within Sakai, There are a limited number of user customizations available:
    a. **Look, feel and Navigation**: Sakai users can customize worksites based upon a selection of a CSS (cascading style sheet) "skins". However, creating these skins is difficult and they must be "pre-developed" by a skilled designer/developer as opposed to an "end-user"/professor.
    b. **Icons:** Icons used in Sakai are not easily configurable and are not part of the CSS layer (right now).
    c. **Tools:** On a tool level, a professor using Sakai can customize which tools he/she wants to include or exclude based upon his or her preferences for a specific course or project very easily.
3. Because of the current diverse course websites in place at UCLA, it is important to recognize Faculty's current ability to customize the look, feel and style of their websites. In Social Sciences, for example, ClassWeb's flexibility allows an instructor to replace the main class page with one of their own design.

8. How does SAKAI support the UCLA distributed operational model?

1. It is difficult to assess this in the absence of a concrete definition of "the UCLA distributed operational model." However, we will assume that this phrase is describing UCLA's environment today.
2. Sakai is a configurable system and runs on many platforms. Additionally, there are two database engines that can be used with Sakai: Oracle and MySql.
3. Sakai also allows each instructor to select the tools that can be used in a course or project basis.
4. Sakai allows for distributed groups to create/code their own "templates" or "skins" for courses that may be independent from any other courses or groups of courses in Sakai. However, this is not simple to do and requires a CCS (Cascading Style Sheets) expert software developer/designer.
5. Sakai therefore supports the UCLA distributed operational model in many aspects, but it is not as flexible as today's current environment where a faculty member can decide to code his/her own class websites and have the Class Web system simply "point to it") as today's current environment.
6. However, Sakai is not like a "shrink-wrapped" piece of software. The system administrator documentation is improving however, integrating Sakai into the UCLA environment is not a simple task. We have done it 3 times but it is complicated.

9. What work and expertise does it take for faculty to develop/locate/add a new tool?

1. The Faculty interface within Sakai allows an instructor to locate and add any tool already installed on the Sakai system to their class/project. This takes no particular technical expertise, though familiarity, and probably some initial training or use of a video tutorial) is likely required.
2. Developing a new tool for Sakai is not something most faculty would be capable of doing as it requires Java expertise and an understanding of Sakai requirements for tools and tool-interoperability.
3. Stuart Sim of Sun is working with Chuck Severance of Sakai to add some tools to Sun's Netbean's development environment that allow java developers to build tools more easily for Sakai.

10. What usability testing results do we get from all types (from inexperienced to very advanced) faculty and student users?

1. Usability is manageable but needs to be refined, even for a very advanced user. There are groups working on this within the National Sakai effort.
2. **Usability testing from 'inexperienced' faculty (i.e. faculty who have very little to no hands-on experience with a CMS) has yielded the following results:**
   Inexperienced instructors regularly get lost trying to do fairly simple tasks (e.g. create/organize Discussion topics, post Resources, add links, and create/organize Assignments). These are tasks that they do not get lost trying to do in WebCT. They require more support than they do using WebCT.
3. **Usability testing from experienced faculty (i.e. faculty who have hands-on experience with at least one tool in a CMS) has yielded the following results:**
   a. Frustration at not having more control over the displays
   b. Annoyance with the HTML editor's abilities (it is outdated)

c. Advanced instructors also get lost doing simple tasks, but are able to figure them out; they are routinely frustrated by inability to set more advanced controls within tools (e.g. setting permissions for students/TAs to view or revise assignments or discussion posts, setting deadlines/selective release for tools for which this is not available, fixing the display order of resources, etc.).

d. They would also like to see in Sakai more seamless integration with campus resources like: library eReserves for their class (including access to music streaming from the Music Library), access to My.UCLA Gradebook for their class, OID's VideoFurnace links for their class, etc.

4. **Usability testing from students has yielded the following results:**

   At least from the Humanities, many students (inexperienced as well as experienced) using Sakai found the UI confusing and not intuitive. This especially applies to accessing/viewing resources, reading/posting to discussions, and gaining access to instructor comments on assignments that have been returned.

11. Does SAKAI provide a sufficiently flexible range of access rights to materials and sites?

1. Yes, Sakai is a very configurable system. It allows the unit that is configuring it to design and implement their own set of roles and access rights. Similarly the access rights of roles can be specific down to the worksite such that one professor could set them differently than another without any impact on each other.
2. In terms of implementation, we would probably implement a standard set of roles and permissions which can be changed by owners of sites to suit their needs. This is because adding a set of characteristics (i.e., defining a new role) involves work by a system administrator.

12. What work is required to take a tool developed for example by the SAKAI SEPP institutions and port it to UCLA? What are the characteristics of tools and of the UCLA infrastructure that make this process more or less difficult?

1. To install tools included in the "Sakai bundle" there is no additional work beyond the basic upgrade work. The new tools get installed when a new version of Sakai is released.
2. To install tools released outside of the "Sakai bundle" we have to do a few things:
   a. Follow institution's installation instructions
   b. Test that new code doesn't disrupt our UCLA- integration code
   c. Test that the new code runs well on our HW and SW configuration

13. Do faculty exploit the double value of the same tool set for teaching and collaborative group work?

1. Humanities faculty piloting Sakai for courses have not also created collaborative group worksites, largely due to shortage of time (busy teaching!), but CDH believes that they see the [3]double value[2] of the tools and how they could be used for collaborations. Said another way: Instructors who have piloted Sakai could easily be convinced to use it for project collaborations in the future.
2. CDH has also attempted to get project workgroups using Sakai, but most have turned to other solutions because of a need for specific functionality (e.g. blogging/wiki, easy organization of

image collections). Sakai should be more useful to them in future versions, with a deeper pool of available tools.

14. In a broad sense, how does SAKAI compare to other Course Management Systems, including proprietary systems. To what extent does SAKAI + available tools cover the primary functions used or requested most typically by UCLA faculty?

1. This is a difficult question since the pilot team as a whole worked exclusively with Sakai and not with the other CMS's on campus. In addition, there is no specific set of requirements that define the "primary functions" requested by UCLA faculty.
2. Sakai is not as mature as many of the commercially available CMS solutions. It is on version 2 while Blackboard is on version ~9
3. According to some UCLA faculty (i.e. Susan Cochran) who have used both Sakai and Moodle or WebCT, Sakai 1.5 was not as 'usable' as either Moodle or WebCT. However, this was data from her experience with Sakai over a year ago and there have been three major releases of Sakai since then.
4. Compared to WebCT Campus Edition 4.x, Sakai 2.0x has one clear advantage in providing a simple individual user file space called MyWorkspace, something not available at all in WebCT). (In WebCT, the MyWebCT page only lists the courses with which the user is associated; it does not provide space for the user to save files on the server.)
5. Otherwise, Humanities users and support personnel generally agree that Sakai feels less intuitive to figure out on one's own, less polished in the way it organizes course content (e.g. cumbersome views in the Discussion tool, inability to define where Resources appear and in what order, etc.) and lacking in controls on some of the tools used most widely by faculty. The primary functions used by Humanities instructors in Sakai are:
    a. Announcements
    b. Discussion board
    c. Schedule (calendar)
    d. Assignments with instructor comments
    e. Drop box (online submission)
    f. display of Student web projects or other digital work
    g. Roster view/download
    h. Gradebook
    i. Email-the-class
6. Email the class functionality is not available within WebCT, but is available within Sakai via the Announcements tool.
7. Gradebook final grade submission is not currently available in either CDH's instance of WebCT or UCLA's Sakai (version 2.01).
8. All of the other tools are available in WebCT, with a much wider and more stable breadth of controls (setting access/permissions, setting times for release/closure of content, specifying display order, etc.) than is true in the current version of Sakai.

15. To what extent and how cleanly does SAKAI implement the relevant standards such as SCORM and OKI?

1. Sakai is not currently OSID-compliant, although there is discussion it may be. From what we have learned at conferences, Sakai developers needed to extend the OSIDs to make Sakai work.

2. Developers at UC Davis are currently working on a SCORM player tool for Sakai, presumably for release with Sakai 2.2 due out this month.
3. Practical implementation of the evolving SCORM technology remains a challenge. For example, the current version of SCORM is "SCORM 2004," which contains the highly-desired "lesson sequencing." The previous version, SCORM 1.2, is no longer maintained or supported by SCORM's defining regulatory body, ADL. IBM's newly-released open source SCORM tools for Sakai, announced at the Vancouver Conference, however, are for version 1.2 and do not work with the current support version of SCORM. Using these IBM tools also necessitates maintaining a separate SCORM Tracking server. While SCORM continues to be a technology worth investigating, adoption of these technologies for Sakai is not yet an easy proposition and real-world implementations in other CMS environments reveal many difficulties (see appendix E).

16. How does the concept of a developing a "common build list" within UCLA, within UC or within SEPP work in practice?

1. The Sakai National community is developing a requirements process that will help to coordinate and bring together the development efforts of the community. This is still in the early stages and it is likely going to take at least six months for the newly formed Sakai Foundation to re-define its work processes and procedures.
2. Sakai also has a "match maker" tool that should help those with development ideas find each other, however it is out of date at this point and has been under-utilized.

APPENDIX A

The Original FCET RFP

APPENDIX B

Original Sakai Pilot Proposal to FCET

APPENDIX C

Sakai Pilot Team Collaboration

Collaborating Departments, the participants and their contributions:

- **The Anderson School of Business**
  - Participants:
    - S. Kumar - Co-Project Manager
    - Howard Kim - Software support, Roles definition
    - Thomas Bush – Java programming - ISIS Integration
    - James Watkin – Java programming - Registrar Data Integration
  - Contributions include:
    - Co-manage the project with ATS
    - pilot testing
    - Provide programming assistance to the project (ISIS & Registrar data integration were completed)

- **The Humanities Division of The College:**
  - Participants
    - Annelie Chapman - Departmental Support, Roles definition
    - Lucy / Harsh Desai - Java programming & Integration code migration
  - Contributions include:
    - pilot testing
    - Created a user feedback survey
    - Provide local support

- **The Social Sciences Division of The College:**
  - Participants
    - Mike Franks – Support – created emailed support form
  - Contributions include:
    - Participate in pilot testing
    - User Support

- **The Office of Information Technology and Academic Technology Services:**
  - Participants:
    - Rose Rocchio - Co-Project Manager
    - Pete Nielsen - Server admin
  - Contributions include:
    - Project Management
    - Participation on the National Sakai - Requirements work group
    - Servers & system admin
    - Technical support

- **The Office of Instructional Development:**
  - Participants:
    - Jim Williamson - Support, Training, Skin
    - Susan Phares – Training videos
    - Slobodan Jovcic (Jovca) – Sys Admin & database help
  - Contributions include:

- Technical & software support
- Participation on the National Sakai - QA work group
- Participation on the National Sakai – UI group
- Training
- UCLA Look (Skin)

APPENDIX D

Sakai National and Pilot Project Milestones

Project Milestones and Accomplishments by Quarter:

Summer 2004

Sakai National - development in progress

Fall 2004

National milestone: Sakai National released v1.0 in October 2004

Winter 2005

National milestone: Sakai National released v1.5 on March 4th, 2005

- Implemented 'Vanilla Sakai', that is without ISIS or Registrar integration
- Project sites:
    - PDP Alumni site – led by Dawn Canfield – implemented mid February – completed on July 15th, 2005

Spring 2005 - began on 3/30/2005

- Began work on ISIS integration
- Maintained & Supported Project sites:
- Pilot Production Course sites:

School	Org	Course	Instructor
The Anderson School	Mgmt	273A Information Systems Mgmt	Jason Frand

Summer 2005 – session C- began on 8/8/2005

National milestone: Sakai National released 2.0 on June 15th, 2005

- Migrated data to 4 new server environments: DEV, TEST, STAGE & PRODUCTION
- Implemented with ISIS – mid summer
- Samigo/"Test & Quizzes" Evaluation sub group – determines that the tool in v2.0 is too buggy for UCLA to use
- Pilot Production Course sites:

School	Org	Course	Instructor
The College	CDH	American Fiction - English 173C	Chris Mott
The College	CDH	Philosophy 31, 05	S. M. Pai

Fall 2005 - began on 9/26/2005

National milestone: Sakai National released 2.0.1 released mid-late summer

- Implemented 2.0.1 with ISIS & Registrar's data on – Late Sept. 2005
- Pilot Production Course sites:

School	Org	Course	Instructor
Anderson	-	Field Study Course – Fall /Winter	
Anderson	-	Field Study teams– Fall /Winter	
The College	CDH	AP/TESL c111/211	Christine Holten
The College	CDH	English Comp 2i	Christine Holten
The College	CDH	French 134	Eric Gans
The College	CDH	French 205a	Eric Gans
The College	CDH	Scandinavian 50	Patrick Wen

Winter 2006 - began on 1/4/2006

National milestone: Sakai National released 2.1 on Dec 1st, 2005
National milestone: Sakai National released 2.1.1 on Feb 14th, 2006

- UCLA Decides 2.1 too buggy to use for Winter Quarter
- OID developed training video's for Sakai using the Breeze technology
- UCLA implemented 2.1.1 in mid February on TEST machine
- Pilot Production Course sites:

School	Org	Course	Instructor
Anderson	-	Field Study Course – Fall /Winter	
Anderson	-	Field Study teams– Fall /Winter	
The College	CDH	06W AP & TESL 101W-1	Linda Jensen
The College	CDH	06W ENGCOMP 2I-1	Christine Holten
The College	CDH	06W ENGL 4W-2	Matt Dubord
The College	CDH	06W SCAND 19-1	Timothy Tangherlini
The College	CDH	06W SCAND 178-1	Timothy Tangherlini
OID	OID	06W HNRS 101E-1	Kumiko Haas

Spring 2006 - began on 3/29/2006

National milestone: Sakai National released 2.1.2 on April 12th, 2005

- Implemented Sakai version 2.1.1 on March 29th – for the start of the spring quarter. This required re-integration with ISIS & Registrar's data
- Created a email-form that allowed students to submit a detailed problem report to the Sakai Support team
- Pilot Production Course sites:

School	Org	Course	Instructor
Anderson	-	Field Study Course-Winter/Spring	
Anderson	-	Field Study teams–Winter/Spring	
The College	CDH	06S AP&TESL 249-1	Lyle Bachman
The College	CDH	06S ART HIS 88S-1	Student Taught seminar
The College	CDH	06S ENGL 88SA-1	Student Taught seminar
The College	CDH	06S ENGL 88SB-1	Student Taught seminar
The College	CDH	06S ENGL 88SC-1	Student Taught seminar
The College	Phys Sci ?	06S ENVIRON 88S-1	Student Taught seminar
The College	SSC	06S HIST 88SA-1	Student Taught seminar
The College	SSC	06S HIST 88SB-1	Student Taught seminar
The College	SSC	06S HIST 88SC-1	Student Taught seminar
The College	SSC	06S HIST 88SD-1	Student Taught seminar
The College	CDH	06S KOREA 187-1	Timothy Tangherlini
The College	Phys Sci ?	06S PATH 88S-1	Student Taught seminar
The College	SSC	06S PHILOS 232-1	Christopher Smeenk
The College	SSC	06S POL SCI 88SA-1	Student Taught seminar
The College	SSC	06S POL SCI 88SB-1	Student Taught seminar
The College	SSC	06S POL SCI 88SC-1	Student Taught seminar
The College	SSC	06S POL SCI 88SD-1	Student Taught seminar
The College	SSC	06S POL SCI 88SE-1	Student Taught seminar
The College	CDH	SCAND C182-1	Timothy Tangherlini

Using Sakai for Collaborative Projects:

The PDP alumni community used the first version of Sakai that we implemented (version 1.5) last spring. Then last fall, The Anderson School used Sakai for their field study collaborative projects creating over 100 project team worksites. Several other departments tried the software or are using it for projects, including Engineering, Campus Security Team, the Sakai project team themselves, etc..

APPENDIX E

SCORM ISSUES

Examples of the difficulty with SCORM Integration:

-----Original Message-----
From: blkbrd-l - A list for Blackboard course administrators and faculty [mailto:BLKBRD-L@asu.edu] On Behalf of Sean Weaston
Sent: 27 January 2006 17:22
To: BLKBRD-L@asu.edu
Subject: Flash Learning Objects - Blackboard 6.3
Hi,

I recently subscribed to this blackboard listserv in hopes of attempting to resolve a problem (or two) that we have been having.

We are attempting to integrate Macromedia Flash Learning Objects into Blackboard as SCORM content. Our goal, which is supposedly possible – as noted by blackboard's website - is to have students navigate interactive flash quizzes/exams/activities and receive a grade that automatically appears in the blackboard gradebook. We have been testing using Flash 8 Pro's quiz template and publishing with SCORM 2004 tracking. We create manifests that are SCORM 2004 compliant and run them via blackboard. Through trial and error we have gotten the content to run on multiple browsers, both Mac and windows. We have experimented with SCORM 1.2 and 1.3 (2004) content, tracking, manifests, and metadata files. I have attempted using wrappers, converters, etc from the ADL website.

What is interesting is that while we were operating under blackboard 6.2, we were able to run flash learning objects via SCORM 1.2 tracking on browsers operating on windows. However, that still alienated many of our users - because most of the learning interactions we develop are for Mac users.

My question is, has anyone been successful in running Flash learning objects on blackboard and getting the grades to report (as actual scores, and not the!) in the gradebook?! If so, we would love some advice.

Thank you,
Sean Weaston
Technology Assistant
Advanced Learning Technologies
Miami University
-----Original Message-----
From: blkbrd-l - A list for Blackboard course administrators and faculty [mailto:BLKBRD-L@asu.edu] On Behalf Of Shen, Dr J.
Sent: Monday, January 30, 2006 1:46 AM
To: BLKBRD-L@asu.edu
Subject: Re: Flash Learning Objects - Blackboard 6.3
Hi Sean,

We have experimented with SCORM contents produced by Macromedia Breeze which contains quiz and our result is similar to yours, i.e. the gradebook only shows '!' not the actual score. The score is recorded in the gradebook and can be accessed by following the '!' trail of clicks. I have asked Macromedia for a standard/demonstration SCORM content to show how it should behave in Blackboard, they cannot oblige. Equally I have asked Blackboard for a standard SCORM content that could demonstrate the correct behavior of a SCORM package in Bb, but they cannot do that either. I have even mentioned this to Volker in person and he does not seem to be clear how should the SCORM score be displayed in Bb. So I don't know whose fault is it and at the same time both sides are reluctant to make any progress. Very frustrated to say the least.

--Jie
-----Original Message-----
From: blkbrd-l - A list for Blackboard course administrators and faculty [mailto:BLKBRD-L@asu.edu] On Behalf of John DiGennaro
Sent: Tuesday, January 31, 2006 7:07 AM
To: BLKBRD-L@asu.edu
Subject: Re: Flash Learning Objects - Blackboard 6.3
From my days in corporate America working with multiple LMS vendors, this typically requires custom API’s to be written for it to 'work.'

Often costly they tended to work providing version of the LMS and the content builder didn't change dramatically, but did occasionally break. Using very standardized process maps, we could create our own content using Breeze and have the results bounced back into the LMS. One caveat, in my 7 plus years of working with projects, this was almost never cross-browser or cross-platform capable.

J