4-9-2008

WILE: A very low cost web-based integrated learning environment

David A. Baker
Bond University

Michael J. Rees
Bond University, Michael_Rees@bond.edu.au

WILE: A Very Low Cost Web-based Integrated Learning Environment

Abstract

Today, Learning Managements Systems (LMS), both commercial and open source, are available to implement web-based learning environments, yet they require large servers, a team of support staff, and have an expensive licensing cost. Open source LMS alternatives often run on leaner machines, but hiring a team to support the system can be costly. The complexity and cost of undertaking the implementation of an LMS can often be taxing on a smaller institution, but they must offer an LMS to compete with other institutions.

WILE (Web-Integrated Learning Environment) is an LMS that uses combinations of either free or low-cost services already available on the Internet. WILE is available anywhere there is an Internet connection through a standard web browser. Fully exploiting the idea of Web 2.0, WILE is a mashup of different services to create a complete web-based environment, different from the wide range of LMS products currently on the market. WILE is a mashup of Google Apps, Google Docs, Google Groups, Gmail, Engrade, ClassMarker, and ProProfs Quiz School. All the required LMS services are encapsulated into a single "web desktop" using Ext JS v2.0.

This paper describes the motivation behind the new approach of building an LMS from near zero-cost web services that resulted in WILE. The Web 2.0 background that inspired the building of WILE is described and the design and implementation discussed. At the end of the development it was found that WILE fulfils 87% of the requirements of a fully functional LMS.

1. Introduction

There is wide range of Learning Management System (LMS) software products available today. The commercial Blackboard/WebCT packages dominate the market. Other options include little-known Angel Learning and open source contenders such as Moodle and Sakai.

An LMS is designed to push content to students; often this process is only one-way. The modular style of current LMS packages allows instructors to track the progress of their students and choose what other features to include. Currently, most LMS packages provide a large range of features and consequently must be installed on large, robust servers, which are often expensive. In addition to the costly servers, commercial packages have significant license requirements which must be renewed regularly. Open source options are more subtle in their costs, but still need highly qualified professional support staff to perform the installation, maintenance, and upgrade functions. The greatest issue for an LMS is continuing maintenance. Over the past year at our institution, our iLearn (Blackboard) LMS occasionally experienced about a day of downtime, where the entire system was unavailable to anyone.

Smaller institutions do not have the luxury of large systems support departments or large budgets and the cost of implementing an LMS can be taxing or even prohibitive. The cost of training instructors how to use the system can be in excess of $20,000 for a group of 20.[HREF1-3] This leads to smaller institutions being hampered by bulky LMS packages with high licensing costs, running on large servers, which require a lot of maintenance staff. Even with open source packages, the above issues still pertain, but zero licensing costs are replaced by expensive specialists and possibly more frequent patching to stay up-to-date.

The aim of the research was to create a learning management system of close to zero cost from components freely available in the public domain. The LMS that resulted is called WILE (Web-Integrated Learning Environment). WILE uses free or low cost services readily available on the Internet and mashes them together to create a feature-rich LMS.

WILE fully exploits the idea of Web 2.0 and pulls together different, disconnect services on the Internet to create a complete web-based application. Student participation and collaboration is an important feature of WILE. Instead of just pushing content to students, WILE fosters discussion by including discussion groups and other collaborative tools. Like the large, expensive commercial LMS packages, WILE is modular and components can be implemented on a class by class basis. Most importantly new services can be easily integrated into WILE in the future. WILE encapsulates all the services into one streamlined interface and differentiates itself from other LMS packages by:

- Requiring minimal development (no need to create the entire system from the ground up)
- Extending web-based services already available
- Not requiring special plug-ins or specific browsers; the browser only needs to display basic HTML with CSS and be JavaScript compatible

At the outset, the research project set an arbitrary cost limit of less than $5,000 per annum, and this was successfully achieved. The low cost allows the WILE to be readily available to schools and universities with small IT budgets. The use of portals and free services made this possible. Institutions using WILE will benefit in the following ways:

- Existing infrastructure available at little or no cost
- Decreased system/network maintenance by local IT staff
- Increased uptime
- Free or low cost class sites, document repositories, email, and calendars

2. Previous Work

LMS packages focus on pushing content to a student and managing the flow of a course. In addition an LMS tracks and reports a students' progress and interactions between the instructor(s) and tutor(s). A modern LMS package requires the course organiser to know little or, more often, no HMTL (J. Redol, D. Simoes, A. Carvalho, et al, 2003). Most of the LMS products in use today are closed-source, commercial solutions. Where the solutions are open source, the amount of specialized support can often match the cost of a commercial system.

Blackboard and WebCT are the two most well known LMS products in existence. In 2006, Blackboard and WebCT merged together [HREF4]. Blackboard is costly to set up and maintain. From the associated documentation a two-server system should be used to run a 15,000 student implementation with 500 concurrent connections and 1,000 courses [HREF5].

Moodle is an open source alternative to Blackboard. Moodle is available under the GNU Public License, and is therefore free. To keep costs low, Moodle suggests the use of Debian GNU Linux with a MySQL database. The downside is support costs: a support team would need to be hired to set up and maintain the operating system and LMS installations. Moodle is very feature rich, but due to the lack of formal support, some schools or universities would have difficulty and higher expenses to implement it on their own campuses. The lower cost of an off-site hosting package might make it more tempting [HREF6].

The Sakai Project is another open source option. Sakai is a bit different, as it has a group of academic institutions to direct development. Massachusetts Institute of Technology (MIT) and Stanford University provide part of the backing for Sakai. Like other enterprise LMS products, Sakai needs to be installed locally, as the only external hosting available is in the Netherlands. Again, the initial setup and hardware maintenance are an issue with regard to cost. Sakai was designed with collaboration on group projects and research in mind [HREF7] .

Blackboard, Moodle, and Sakai provide an idea of an acceptable framework and the general framework of a fully functional LMS. It should be a portal-style environment and have a wide variety of features, so it is useful to many different types of classes including options for student collaboration.

Commercial LMS products have a regular license cost to continue using the system. The benefit of open source alternatives are the lack of license cost, yet the costs associated with open source systems are those of proper installation and setup of the server(s). Indiana University spent over $1 million US on installation, setup, and updates to Sakai. Although they saved in license costs, the amount consisted "mostly of existing developers and functional specialists staff salaries" [HREF8].

Some universities might consider using external hosting to help lessen the burden of an open source LMS. Moodle offers fee-based shared hosting option [HREF6]. Costs for an external host range from $3,000 USD for a basic package for a 3,000 student/100 course campus (about the population of Bond University) to over $4,500 USD for the more suitable packages [HREF9, 10].

Blackboard is usually installed on servers located at the university. Blackboard recommends a two-server setup: Web Site/Application and Database. Blackboard assumes a small institution is: 1,000-5,000 users, 500 concurrent users, 500 courses, 5,000 enrolments [HREF11].

Dell servers which meet Blackboard's specifications are about AUD $4,222 each. This is before adding additional CALs, Microsoft SQL Server database multi-processor licenses, and a support contract in the event of a server crashing. Bond University uses six servers to serve the entire student and staff population at the university, providing an idea of the server costs.

The main concern when hosting onsite is server maintenance. A server which is having issues or crashes can cause part of all of the LMS to be unavailable. Three options are available: retain staff to maintain the servers, purchase a support contract from the server vendor, or a combination of both. Usually an institution will hire employees for routine maintenance and sign a support contract with the server vendor, for major issues.

Recent years have brought a number of advances in how the Internet operates and how people interact with it. One concept is Web 2.0, which introduces the idea of using the web as a platform, much like an operating system is to a computer. Web 2.0 is about providing continually updated data to the end user, and an area for collaboration between users [HREF12]. The term Web 2.0 was first coined by Tim O'Reilly in 2005 [HREF13]. Through three case studies, he determined three things that separate a Web 2.0 application from a Web 1.0 website:

1. **Web Platform v Desktop**. In the comparison of Netscape with Google, O'Reilly noted that while Netscape had focused on a desktop application, Google used the web as an application and created a database backend for massive data access.
2. **Reach More Customers**. When comparing DoubleClick with (Yahoo) Overture and (Google) AdSense, O'Reilly pointed out that DoubleClick only has about 2000 customers, while each of the other services serve hundreds of thousands of clients. Although DoubleClick was one of the first Web 2.0-style applications (as it integrated its content into another site), they focused on just a core number of sites, whereas Yahoo and Google focused on making their ads more "customer friendly" instead of using popup-style ads.
3. **More Users = Better Service**. Akamai focused high-demand sites and received revenue from them, but BitTorrent uses each computer that joins as a hub to distribute content. When the load became too high, Akamai has to add a new server, where a BitTorrent network grows with the number of users[HREF13].

From these comparisons, it was determined that Web 2.0 applications/sites need to use the web as a platform, have information that users are looking for, be customer-friendly, and be flexible to handle high demand. Ebner and Anderson (2007) suggest Web 2.0 appears in a number of distinct areas:

- Blogs / Weblogs
- Wikis
- Podcasting (Audio Blogging)
- Web Sharing Applications
  - Social Networking Websites
  - Tagging/Social Bookmarking
  - Multimedia Sharing
- RSS/Aggregation Services [HREF14](Ebner, 2007)

The above services model the idea; yet do not fully exploit Web 2.0. These services do use the web as a platform, but simply replace the desktop (or magazine for advertising) with the web browser. To fully implement the idea of Web 2.0, there should be a communication between services on the Internet combined to create an interactive environment, rather than just pushing the information to the end user. Three of the areas above are interactive environments. Blogs allow users to respond and communicate with the blog author and the author to respond back, Wikis use web services to create an interactive mini-program, and Web Sharing Applications involve sharing information or communicating with other subscribers to the service.

E-Learning 2.0 is a new buzzword for a new way of teaching students. Teachers had been teaching the same way for decades, they gave a lecture in front of a class, handed out materials, gave assignments and announcements all in class, during the designated time. With the advent of the LMS, instructors were able to post the material online, along with assignment and announcements. Teaching did not change much, there were still lectures for class, and students simply printed out material and brought it to class (Ebner, 2007).

The advent of Web 2.0 changed how people interact online. Collaboration is an important aspect of E-Learning 2.0. Brown mentions that Web 2.0 technologies should be integrated into an LMS, where appropriate [HREF15]. Ebner mentioned that Web 2.0 is as much a social idea as it is a way to utilize the Internet. Therefore, the rate of uptake of E-Learning 2.0 is as much or more a human-related factor as technological (Ebner, 2007). Students are already blogging and using wikis (as in Wikipedia). Beginning to add them to an LMS is a logical step, so they can comment on a topic in the class. Instead of weekly writing assignments, students can create blogs.

Another technology that has come to prominence is the mashup. A mashup is a combining of services into a new service. Part of mashups are web services, these provide the data for the mashups to utilise. Mashups are "ostensibly hybrid applications, where two or more technologies or services are conflated into a completely new, novel service" [HREF16]. Floyd et al noted, "The original vision of the web was of a system for academics to share information and data in the form of documents." The idea of a mashup is that it fits into the rapid development model. Existing services can be pulled together—"mashed up"—for a different final product. Developers are quickly able to see results.

The design team should go through five steps:

1. Determine a rough idea about how the new system might look and function
2. Select services which together would be close to the functionality needed
3. Mashup the services into a rough prototype
4. Deploy and get feedback from users
5. Review how feedback relates to the prototype, and quickly repeat [HREF17]

While teaching a class, Floyd et al saw that people would develop mashups to complete problems in the real world. One example was a campus route planner for new students on campus. What the authors concluded was that mashups and patchwork prototyping worked well because:

- Users could created their own application, without experienced programmers; so,

- It involved the end-user in the development process; which,
- Allowed for a better product that could be integrated into the end-user's everyday life [HREF17]

Current systems provide a good base of requirements and usability of the environment of an LMS. While not all features were implemented in WILE, a good set will be derived in part from these systems.

Also, Web 2.0 is changing how people use the Internet. The Internet has changed into a social platform instead of solely a place to exchange information. The advent of blogs, wikis, podcasting, and similar technologies has given the users of the Internet more ways to communicate and interact with each other. Using these new technologies it is possible to move from the decades-old methods of teaching and utilize a concept of E-Learning 2.0. This recognizes that the learning environment should be a social one. It should assist in increasing interaction between students which in turn aids the learning process in that they apply what they learn.

E-Learning 2.0 can use mashups to assimilate information from various services on the Internet. These mashups should be small, easily created interfaces, which are also convenient to use and implement. Lastly, web services can be used to provide data to the mashups. Most importantly, above all, E-Learning 2.0 mashups must be platform independent.

3. Web-Integrated Learning Environment (WILE)

WILE is a mashup which incorporates the important and/or most used aspects of a Blackboard-like environment, including: assignment listing and submissions, discussion forums, a gradebook, and more. A central portal is the best design for WILE, which uses the concepts of Web 2.0 and transfers them to an LMS model, without making the environment more difficult to use. A mashup for WILE meant finding technologies which link together in a useful manner.

Students are multitasking more often. They expect an interactive environment to express their ideas and work out problems by talking it through. With the current LMS formats available, students are told what to think, and there is very little engagement online. Since the LMS is where most students spend a good amount of time, it would be good to integrate concepts of Web 2.0 with which they have become familiar.

A desktop style interface is something that computer users have become accustomed to. Students and instructors are familiar with navigating through different windows and multitasking. Yet, when it comes to current LMS packages, content is displayed as a hierarchy of pages. To view the syllabus, assignment list, and files to download for the project, one would have to go back and forth in the pages for the course. Alternatively, one could open each window in a tab or new widow, but that just clutters the taskbar and desktop.

In addition to working on assignments, many students also have open their choice of MP3 player, instant messenger (with multiple conversations), and possibly email and/or a Microsoft Office program open. Quickly the taskbar becomes too small creating a need for multiple clicks to get back and forth between an assignment and the LMS. Instead WILE considered an option that combined all of the learning content and associated windows encapsulated into one browser window. WILE allows users to free their taskbar of clutter and keep the student in one environment, from the reading of assignment tasks to completing and submitting them.

3.1 WILE Features

While most examples of an LMS push content to students like a textbook, WILE uses familiar environments and tools so students can become involved in the discussions. WILE focuses on including reusable content and allows for additional features to be easily added later.

The University of Texas conducted a survey of which features are best included in an LMS. The study focused on the tools from an instructor standpoint. They found that the most frequently used features were [HREF18]:

1. Syllabus
2. Content
3. Contact Details
4. HTML files added
5. Announcements
6. PowerPoint presentations
7. Gradebook
8. Email
9. Discussion Board
10. Word (documents uploaded)

The Educase Center for Applied Research (ECAR) released a study in September 2007 regarding "undergraduate students' use of and preferences and expectation of IT" [HREF19]. The ECAR study included a section on CMS Use and Experience, which outlined what students regarded as the most useful features (with average percent usefulness) [HREF19]:

1. Keeping track of grades on assignments and tests online (87.6%)
2. Only access to sample exams and quizzes for learning purposes (83.4%)
3. Online Syllabus (79.6%)
4. Turning in assignments online (76.4%)
5. Online readings and links to other text-based course materials (76.2%)
6. Getting assignments back online from instructors with comments/grades (74.8%)
7. Taking exams and quizzes online for grading purposes (73.2%)
8. Online sharing of materials among students (70%)
9. Online discussion board (62.6%)

Based on the results of these two studies, the syllabus is a must, followed by a gradebook and discussion board. The teachers seemed focused on pushing content to students, where the students were looking for features that would allow them to track and increase their performance in a subject.

Using WILE, students can work on their assignments from any location, regardless of the software installed on the computer. For instance, they could use Internet-only terminals as only a modern web browser is required to access the software. Completing assignments online is a unique option as it allows students to complete assignments within the same interface they submit the assignments.

The set of features chosen for WILE are:

1. Announcements
2. Assignment Drop Box
3. Assignment Listing
4. Syllabus Page
5. Instructional Content Section
6. Notebook
7. Complete Assignments Online
8. Class Schedule
9. Link Share
10. Discussion Forum
11. Gradebook
12. Quizzes

One last area to note is user access and security of the site. While these services need to be low cost or free, they must be robust enough to limit access to the appropriate persons only. For example, while the instructor and tutors may be able to update content and see the grades for all the students, each student should only be able to view content and see their grade.

3.2 WILE Implementation Platforms

Three main platforms (Yahoo, Windows Live, & Google) where chosen for evaluation. Each service has specific aspects which make it useful, but preferably only one service should be used. This allows the linking of the other services to be more easily integrated. A single service acts as the main portal.

Google provides APIs to add and integrate services and functionality allowing users to host their applications on Google servers, but lacks some of the privacy, needed for students. Yahoo provides access to their portal via a framework for developers, yet a separate host needs to be found to run the services created. Each portal also has a separate proprietary authentication system, which is a reason to try and stay with one provider during the course of the research.

The Yahoo developers' network has a larger variety of languages for which it supplies information and support. Each language has its own Developer Centre, where examples and documentation on how to interface are provided. The Developer Network also provides links back to the language vendors for additional information. The most prominent offerings from Yahoo are the Pipes and Widgets services. Widgets are desktop applications that connect to a web service for information. Pipes, a new service from Yahoo, are mashups. They allow users to aggregate data from various sources and provide all of the data through a single interface/point. Pipes can be accessed by Widgets [HREF20].

The Yahoo User Interface (YUI) comprises a collection of development resources in JavaScript and CSS that allow developers to create rich web applications. YUI is linked into Yahoo services: Yahoo home page, mail, Flickr, etc. It is an open source solution that is based on the BSD license. Developers can use YUI as a base or supplement for a website. Because YUI uses AJAX and DHTML for the scripting, the resources are cross-platform and cross-browser. YUI extends the feeling and flow of the desktop to web applications [HREF21].

Windows Live is a collection of personal services that Microsoft is offering for free, in most cases. The Windows Live SDK allows developers to build applications and mashups using Windows Live services. There are two main options which would be suitable for creating a mashup using Windows Live services.

Live Spaces is Microsoft's free social networking platform. Originally debuted in 2004 as MSN Spaces, it is an alternative to MySpace and Yahoo's 360 degrees [HREF22]. Users receive a number of tools for them to use in their space, including: Web Logs, Photo Albums, E-mail Uploads, Music favourites and Lists, Contact Cards, Gleaming, Permissions [HREF23].

Popfly is a visual mashup builder based on Silverlight from Microsoft. Popfly Creator is the online visual tool where users build in a web browser, without having to download any more software. Popfly Explorer is a downloadable add-on for Visual Studio or Visual Web Developer Express; allowing users to create anything from a gadget, to a mashup, or a full web page. Popfly Space is 25MB of free hosting, per user, from Microsoft, for Popfly applications [HREF24-26].

While it has a number of good services, Windows Live is lacking in what is currently available. Live Spaces seems more of a social networking tool. One of the main deficiencies is the lack of a discussion forum. Also, online assignment completion is not included, but there could be integration with Microsoft Office products.

The world's largest search engine [HREF27] is consistently adding new services. Some of the features outlined above are available from Google, yet will need to be "glued" together to make a more coherent interface. Add-ons can fill out the areas where Google may be deficient.

Google Page Creator (GPC) is a service from Google which allow users to create basic web page or full web sites. Each account is given a sub-domain and is hosted for free by Google. Each site is given 100MB of storage, almost twice as much as Windows Live Spaces, which only includes a total of 55MB (25MB for Popfly, and 30MB for photos). Users update the pages in a WYSIWYG interface, with the ability of viewing the HTML also included. While GPC does not officially support JavaScript for more advanced functions, the help files point users to locations where others have succeeded. FTP is not supported.

Google Groups (http://groups.google.com) is another service that allows users to create and manage their own groups complete with group-wide mailing lists. Google Groups allow group owners to determine access levels and privileges of members. Integrated Google pages, common file repository, and discussion groups nicely round out the service.

Google Docs (previously Docs & Spreadsheets) is a free online productivity suite. Currently it includes Documents, Spreadsheets, and Presentations. These three document types are competition for Microsoft Word, Excel, and PowerPoint applications. Google Docs allows users to import those three types of files, and will convert them to the appropriate Google Docs format. Export options to Microsoft Office, OpenOffice, and PDF are all supported. Presentations can now be embedded into external web pages.

Google Calendar allows users to order and manage appointments and meetings. Guests can be invited and if they are also using Google Calendar, they can add the meeting directly into it. Multiple calendars can be made, but viewed in a combined mode. One of the most novel features is the notification options: Popup, Email, and SMS. Settings can be set to display a certain number of minutes before an event. Also new, changed, deleted, and accepted invitations can go to Email or SMS. Finally, a daily email can be sent with a user's agenda.

Recently, Google starting offering a program called Google Apps. There are three levels Standard (free), Premium/Enterprise, and Educational. Google Apps bundles a number of the features previously mentioned:

- Start Page – A common Google-style web page that can have the organizations logo included.
- Chat – Accesses using downloadable Google Talk.
- Web Pages – Integrated Google Page Creator for a company web site.
- Email – uses GMail, but can be branded with the company's logo.
- Calendar – Each user can have multiple personal calendars as well as access to the group/organization calendar.
- Docs – Each user has their own Google Docs Account.

The Google Apps are a very useful combination of services. One of the best options is that a user can select their own domain. They can either pay Google to register it for them or find their own registrar. Google provides specific setup instructions for preferred registrars, but has general instructions for those who host there on DNS servers.

3.3 WILE Desktop Environment

To improve continuity when using WILE, the services should be encapsulated into a common interface. While all of the features could be accessed and grouped by using a tabbed browser, it can often be difficult to mange all the tabs. Furthermore, if a web-only terminal is used (such as the email stations in the Bond University Main Library), having multiple windows are hard to navigate, as the systems are designed to only have one browser window running at a time.

Ext JS is a framework that allows developers to use AJAX without knowing a significant amount of JavaScript. . Ext JS is compatible with:

- Internet Explorer 6+

- Firefox 1.5+ (PC, Mac)
- Safari 2+
- Opera 9+ (PC, Mac)

Cross-browser compatibility is supported by an extensive framework which can be extended. Another benefit of Ext JS is that it is platform and language agnostic. While the client code must be in JavaScript, the back-end and rest of the coding can be done in any language: .NET, PHP, Java, etc.

Ext JS v2.0 allows developers to create AJAX-enabled web sites without having to worry about programming for browser tricks. There are a number of pre-built tools available:

- Lightweight HTML editor
- Scrolling tabs (Similar to when IE or Firefox has too many tabs)
- Improved Tree View (Column Tree)
- Web Desktop

The Web Desktop is the main reason why Ext JS was chosen. The Web Desktop allows the encapsulation of all the WILE service windows in a separate environment, the WILE desktop. This neatly isolates WILE from the rest of the system. Best of all, there is nothing to install! The WILE Desktop will work on web-only terminals, provided they meet the minimum browser requirements of conforming to JavaScript standards.

While each service has a useful benefit, the inability of Yahoo to host content created within its services makes it an unlikely option. Microsoft has done a better job, but has created a Facebook or MySpace style of environment. Each space focuses on a person, not necessarily the content. For instance, an instructor teaching a couple of courses would have all the information on the website, for all the students able to view, their permissions only apply to the profile as a whole.

Google on the other hand has many of the features covered. Below is a comparison of the features list and the Google Applications which could fulfil that need:

- Announcements – Google Groups (Pages or Files)
- Assignment Drop Box – Give Instructor per File access in Google Docs
- Assignment Listing – Google Groups (Pages or Files), also Google Calendar
- Syllabus Page – Google Groups (Pages or Files)
- Instructional Content Section – Google Groups (Files)
- Notebook – Downloadable Google Notebook browser add-in
- Complete Assignments Online – Google Docs
- Class Schedule – Google Groups (Pages or Files), also Google Calendar
- Link Share – Google Groups (Pages, links to outside sites on page)
- Discussion Forum – Google Groups (Discussion Forum)
- Gradebook – Not covered by Google. Will need another add-on service.
- Quizzes – Not covered by Google. Will need another add-on service.

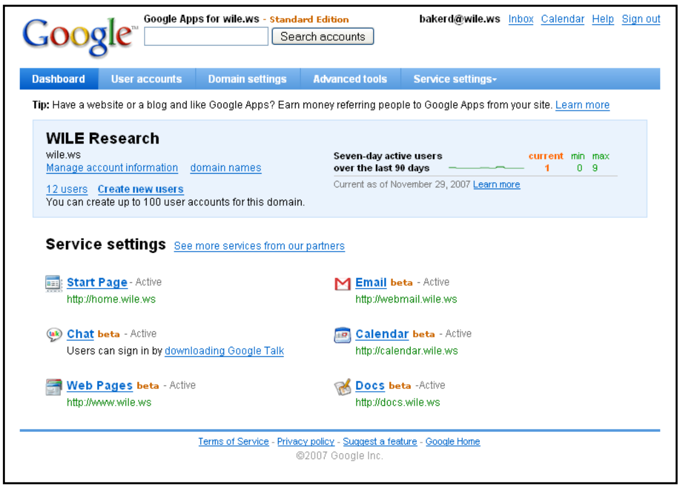
As shown above, most of these features are covered by services from Google. Thus Google was adopted at the main WILE platform with a small number of features provided by other platforms. For example, a non-Google option for the Notebook and Link Share options would be Del.icio.us. On top of Google services Ext JS was used to fully encapsulate them into a familiar environment.

4. WILE Implementation

Google Apps allows users to create an address with them, but the domain extensions are limited: com, net, org, info, and biz. In the example here, the .ws extension is used. Since .ws is not one of the domain extensions offered, it was purchased from a third party provider then configured in the Google Apps Dashboard (Figure 1).

Google Apps provides the users with their own separate email addresses, along with over 5GB of space per email address. Checking with the Google Apps feature comparison page, the amount is consistently being updated in real time, without page refreshes.

In Gmail, Google decided to use a feature called Labels. Multiple labels can be



Tip: Have a website or a blog and like Google Apps? Earn money referring people to Google Apps from your site. Learn more

WILE Research
wile.ws
Manage account information   domain names

12 users   Create new users
You can create up to 100 user accounts for this domain.

Seven-day active users over the last 90 days	current	min	max
	1	0	9

Current as of November 29, 2007 Learn more

Service settings   See more services from our partners

Start Page - Active
http://home.wile.ws

Email beta - Active
http://webmail.wile.ws

Chat beta - Active
Users can sign in by downloading Google Talk

Calendar beta - Active
http://calendar.wile.ws

Web Pages beta - Active
http://www.wile.ws

Docs beta -Active
http://docs.wile.ws

Terms of Service - Privacy policy - Suggest a feature - Google Home
©2007 Google Inc.

Figure 1 - Google Apps Dashboard for WILE.ws domain

created and a message may have multiple labels, being useful for students, as a particular message may not fit neatly into one folder. For example, messages for a particular class project might be better saved under the course and the project name. By default, all messages are shown in, but a user can choose to view messages with a particular label. Google also automatically notices when a message is a reply, displaying the messages in a thread-like view, so the user can view previous messages on the same topic, instead of searching for the related message(s).

Google Docs is a feature which is just over a year old. The technology was acquired when Google bought the Writely start-up. They have steadily added new features and applications over the past year. Currently, the system included three programs which directly compete with the Microsoft Office suite. Figure 2 shows the File Manager interface for Google Docs.

Documents was the first product available. The Word-like competitor allows users to create new or upload document files. File formats accepted by the upload function include: HTML, Microsoft Word docuements, Rich Text, OpenDocuemnt Text, and StarOffice. The maximum file is is limited to 500KB.

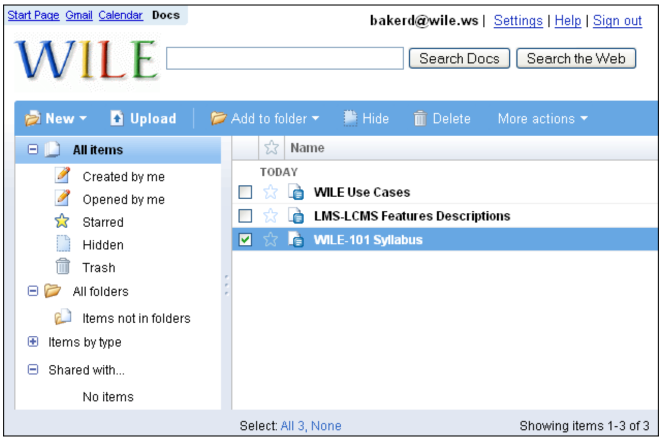


Figure 2 - Google Docs File Manager

Spreadsheets is the Excel lookalike. While users cannot create pivot tables, many common functions use the same name as Excel. Uploads accepted for conversion are: CSV, Microsoft Excel, and OpenDocument Spreadsheet.

Presentations is the newest feature, only about a month old. I creates PowerPoint-like presentations, which can be viewed in a web browser without any special plugins. Importing PowerPoint presentations are allowed from both PPT and PPS formats.

Students can use Google Docs to complete assignments online. Students would share the documents with their tutor or instructor for them to review and mark. Since Google automatically saves versions of the file, the student would be able to view what they originally submitted, if there were to be any questions.

Collaboration is an important feature in Google Docs. Students can simultaniously work on the same document, up to a limit of 50 students. Individual changes are merged into the master document. The collaboration feature is also available even if a student downloads the document, works with it offline, and uploads it later. The offline changes would be merged into the document. Currently, students share a Word document that they pass to each group member to insert their section. Alternatively, all the group members would have to email their sections to one person, who would put the document together. The same feature is provided to Google Spreadsheets, with up to 50 simultaneous collaborators working on the same spreadsheet.

The final Google Apps tool is the Calendar function, which is extremely useful. It allows users to each create their own calendars (yes, multiple ones). A student could create a calendar for classes, social events, and work. Each class and club could have their own colour too. Instructors can create a calendar for each class and then share it with students. Students can then record all the class times to their own calendar(s). Also, the instructor can post a class calendar on the internet, allowing anyone to view it. The interface to add events and meetings is very easy to navigate. In the case of meetings, other users can be invited and can add the meeting to their own calendar when they accept.

Google Groups is used as the central repository for class information. Each class would have a separate group. Group access and ability to view pages would be restricted to invitation-only, keeping spammers out of the group. The group name would be similar to the course code used by the university. To keep groups created by the university together, a common prefix should be used for all the university's classes. For instance, Bond University could use BOND [4 character code]-[3 digit course number] such as BOND INFT-111. While BU would seem a more logical choice, BU stands for Boston University in the United States, and groups are international.

The groups allow members to have varying roles. The individual who creates the group is the owner. Owners can share their access level with others in the group (managers), allowing tutors to add content and monitor discussion threads. All others users are members, who can also be assigned special privileges.

- Each group contains the following five sections:
- Group home page
- Group discussions
- Group members list

- Group pages
- Group files

Each section of the group can be put into a different order or be renamed. For the WILE-101 group, the members section was renamed Classmates, Pages expanded to Web Pages, and Files modified to Course Content. The modified names assist users of the group to have a better undestanding of the content of each section. Home and Discussions were descriptive enough. The following page includes some images of the WILE-101 group.

Even though Google was able to supply a good portion of the tools needed for WILE, it is still deficient in areas specifically related to education: quiz and testing tool and grade books. Additional external services must be integrated into WILE to support these features.

ClassMarker (http://www.ClassMarker.com) is a free, ad-supported online quiz tool. It assesses and grades students using three types of tests (quizzes): Multiple choices, Fill-in, Capitalization and Punctuation. The tests cannot currently be mixed, but new features are consistently being added. The free version comes with advertisements, but for USD $25.00/year the instructor and their students can enjoy an ad-free environment.

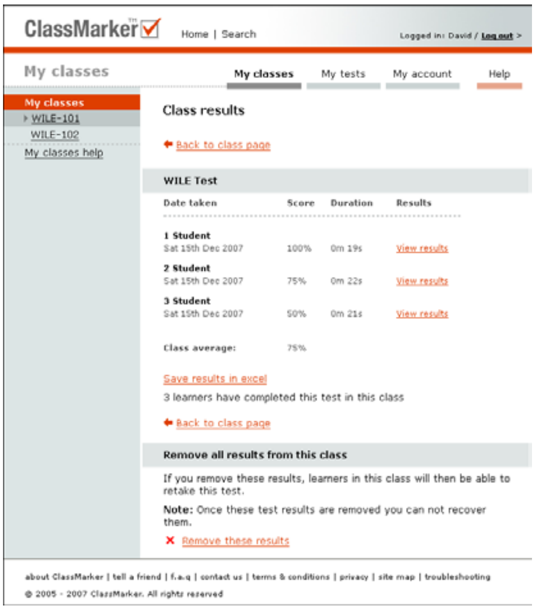
Each instructor is limited to 1000 registered students, and they must de/allocate each space in each class. Registration is done one of two ways: the instructor adds each student or hands out auto-generated registration codes allowing the students to register themselves. Unfortunately, the system does not recognize when the student has the same email address. Each student/email address can only be assigned to one class. For example, Student 1 was registered for WILE-101, upon registering Student 1 for WILE-102 it removed the student’s access to WILE-101. So, each student should self-register for each course, causing students to have a different username for each class. While having a different name for each class is not preferred, the system is still in development. Instructors are able to have multiple classes.

There are a number of beneficial features already included:

- Results Shown.
- Practice Mode
- Randomize Questions
- Randomize Answers
- Email Learner Results

The most useful features are the Randomize options. Random questions and answers mean that an actual test can be used for practice tests, as the test does not have to be repeated and students cannot simply remember the question numbers and answers.

ClassMarker allows users the option of being able to embed the quizzes into their own web site. Also, instructors can create publicly accessible quizzes, but currently ClassMarker does not save the scores of those tests. Figure 3 shows a class’ results for a particular quiz. Instructors can also drill-down to see the actual results of each individual student. While ClassMarker is a bit limited in its abilities, it appears that it will begin to be steadily developed into a very powerful quiz service.



Class results

Back to class page

WILE Test

Date taken	Score	Duration	Results
1 Student Sat 15th Dec 2007	100%	0m 19s	View results
2 Student Sat 15th Dec 2007	75%	0m 22s	View results
3 Student Sat 15th Dec 2007	50%	0m 21s	View results
Class average:	75%		

Save results in excel

3 learners have completed this test in this class

Back to class page

Remove all results from this class

If you remove these results, learners in this class will then be able to retake this test.

Note: Once these test results are removed you can not recover them.

✕ Remove these results

about ClassMarker | tell a friend | f.a.q | contact us | terms & conditions | privacy | site map | troubleshooting

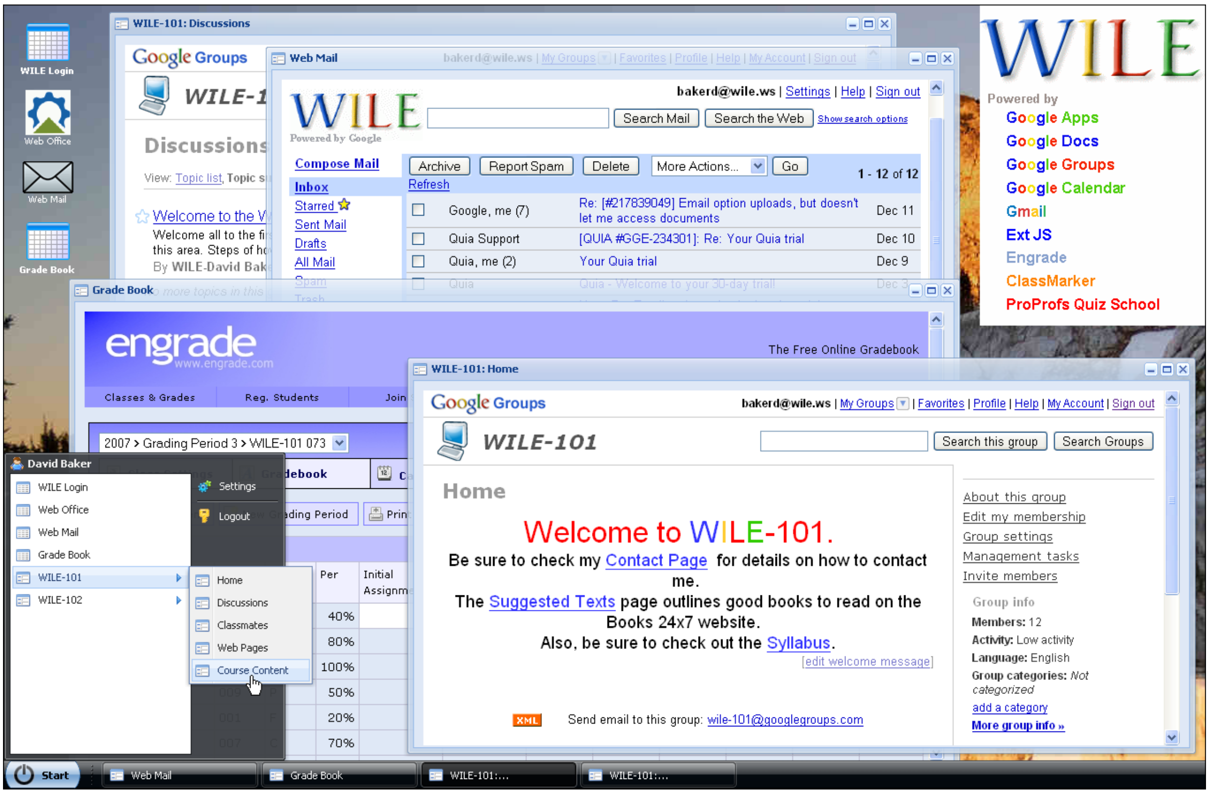
© 2005 - 2007 ClassMarker. All rights reserved

Figure 3 - Quiz Results Page

Engrade (http://www.engrade.com) is a free gradebook. The entire application contains no advertisements. Each school year can have up to nine periods. The instructor can assign a class name, school year, and grading period/term. Engrade includes preset grading scales, but a custom scale can be created. Grades can be either be calculated automatically using the selected scale, with or without weights, or can be assigned manually. Eight categories are available to group assessments. Each category can have a weight percentage associated with it. The new calendar feature can easily have entries added to it. Instructors can print off student reports, either individually or together.

Engrade allows students to log into their account to view their grades and has also taken into account that a student may have more than one ID, as they may have one for each class. All the IDs can be associated with one account. Student IDs can also be added after the account has been created. Engrade proved to be the perfect option for a gradebook. It was developed with simplicity in mind, but allowed to be very versatile and allow complex options.

Figure 4 is a screen capture of WILE in action. The interface is very familiar, as it emulates a desktop experience. The current logged on user's name is show at the top of the Start menu. Each service is easily accessible via a task bar across the bottom of the screen. The taskbar colours and background can be changed to meet a university's preferences, too.



Google, me (7) — Re: [#217839049] Email option uploads, but doesn't let me access documents — Dec 11

Quia Support — [QUIA #GGE-234301]: Re: Your Quia trial — Dec 10

Quia, me (2) — Your Quia trial — Dec 9

Quia — Quia - Welcome to your 30-day trial! — Dec 9

Welcome to WILE-101.

Be sure to check my Contact Page for details on how to contact me.

The Suggested Texts page outlines good books to read on the Books 24x7 website.

Also, be sure to check out the Syllabus.

Send email to this group: wile-101@googlegroups.com

Figure 4 - WILE Web Desktop Environment

5. Research Outcomes

The research and creation of WILE was confined to a thirteen-week period. Due to the shortness of the timeframe, some features needed to have workarounds applied to make WILE functional. Additional time would allow the system to become more fluid.

WILE was able to fulfil about 87% of the features outlined earlier. Below is a list of features, grouped into how fully they were implemented:

Fully Implemented (67%)

- Announcements
- Syllabus Page
- Class Schedule
- Notebook
- Link Share
- Discussion Forum
- Quizzes
- Gradebook

Partially Implemented (18%)

- Assignment Listing (50%) – Due to needing to enter the information in two locations: Google Groups Pages and Google Calendar.
- Instructional Content Section (80%) – Due to size limitation in Google Groups.
- Complete Assignments Online (90%) – Due to lack of viewing formatting for a printed page.

Implemented Using Workarounds (2%)

- Assignment Drop Box (25%) – Due to needing to add the instructor as an editor and needing to send an email to instructor, indicating the assignment is complete.

Fully implemented items are considered to be at an acceptable level consistent with at least what is currently available in major LMS products. The need to log into different services was not a consideration for the implementation of the feature.

With regard to the partially implemented section, the Assignment Listing feature was able to be implemented in a basic form. The Assignments due dates would be listed on a Google Groups Page, but the due date would need to be separately added to the calendar. Students would then have to note the new item on the calendar and add it to their personal calendar. It could be possible to automate the addition of an assignment by utilizing APIs provided by Google.

The Instructional Content Section would be sufficient for many classes, but was noted as partially completed as the 100MB provided by Google Groups would be too small for classes such as graphic design or even law classes, which tend to have more downloads available to their students. This could be corrected with some additional time and programming.

The Complete Assignments Online requirement was met, for the most part, but it would be better if the final layout of a document could be seen while typing. Also, because of how the Google Docs website is created, it takes over the WILE environment and opens new tabs/windows for each document opened.

Finally, the Assignment Drop Box was considered to be a workaround, as there is no central drop box for assignments. Students must first add the instructor or tutor as an editor to the document then email the instructor/tutor to let them know the assignment was completed. Comments from the instructor/tutor would need to be added into the text of the document in a different colour. The originally submitted document is stored in the revisions saved for the document.

6. Further Research

WILE was designed to see if it were possible to create an online environment for learning. Additional research opportunities abound in this topic.

At the moment for each service to be used, a separate username and password must be remembered. There are some options available to minimize the number of passwords for instructors and students to remember: CardSpace, Open ID, and proprietary systems.

CardSpace is a technology from Microsoft which allows users to have a virtual card for each website login. Cards can also be issued from trusted sources, such as a university. If the institution were to host its own OpenID server, they could guarantee all users were from the university. It is important for the university to host its own OpenID server because of the decentralized nature of the system. Google and CardSpace are both going to or already support OpenID. Users do not have to install anything on their local computer to use the authentication mechanism. Finally, Google, Yahoo, or Windows Live could integrate their logins with the other services, creating a single sign-on. Google and Windows Live have this feature for all of their services. Google can also be used for authentication on other websites. Microsoft offers this service also with the Passport/Windows Live ID system.

Two issues were encountered regarding storage:

1. Where is information stored and served from?
2. Where is the information stored long term?

There are several options for the storage and serving of data: self hosting, shared hosting account, or dedicated servers. Self hosting would not be as demanding when using WILE as compared with some other LMS products. WILE will work on any platform, as it is just serving HTML web pages with JavaScript to make it function like a desktop. A shared or dedicated server would be a better option, as theses servers are kept up-to-date with patches and the whole point of the company is hosting websites. This would put the burden on the hosting company rather than the IT department, and without too much cost.

Amazon Web Services has an option to use its Elastic Compute Cloud (EC2)[HREF28] to have a server running on their systems. This would be a fairly small server, but WILE is not processor-intensive. Also, Amazon could take care of both short- and long-term storage issues with its Simple Storage Service (S3)[HREF29]. S3 allows users to have multiple “buckets” where they can store up to 5GB in each bucket. Users only pay as small amount for the space they actually use, not what they are allocating for future use.

Lastly, whereas WILE works well at pushing content back and forth between instructors and students, it does not have any options for real-time contact. Included in Google apps is a chat client called Google Talk. Each person who has an account in Google Groups also has an account that can be used in Google talk. Gmail includes the functionality of chatting with other Gmail users without requiring a download. If functions such as groups chatting are needed, this would require a download. Students would be able to chat with instructors and save the chats through Gmail and still able to access WILE from anywhere, but would need to use a computer with Google Talk installed to participate in group chats. Another program which incorporates multiple chat services is Trillian. Trillian allows users to sign into a single client and have access to chat with all of their friends across various services. Again, this would require a program installed on the client computer.

The WILE prototype was able to support about 87% of the features required for a fully-fledged LMS by the conclusion of the project. This result exceeded expectations. While newer Web 2.0 services might be expected to address some of the remaining 13% of features it is expected that some additional mashup software development will be needed to reach the 100%. Nevertheless it would seem that the WILE approach might lead to a working LMS suitable for at least small to medium scale institutions. This seems achievable for a budget at least one order of magnitude less than that being expended on the open source Sakai project for example. It has to be admitted though additional user testing and a teaching pilot will be needed before WILE is fully proven.

References

J. Redol, D. Simoes, A. Carvalho et al. (2003). "VIANET — A New Web Framework for Distance Learning", The 3rd IEEE International Conference on Advanced Learning Technologies, Athens, Greece, 2003, pp. 258-260.

M. Ebner (2007). "E-Learning 2.0 = e-Learning 1.0 + Web 2.0?", Second International Conference on Availability, Reliability and Security, Barcelona , Spain, 2007, pp. 1235-1239.

Hypertext References

HREF1 Blackboard Learning System™ – Enterprise License: Teaching and Learning Online Course Schedule
http://blackboard.com/resources/training/BLS_Online_Schedule_Summer_2007_FINAL.pdf
HREF2 Training for the Individual
http://www.moodlerooms.com/training-for-the-individual.html
HREF3 Training for the Organization
http://www.moodlerooms.com/training-for-the-organization.html
HREF4 Blackboard Inc. Completes Merger With WebCT, Inc.
http://investor.blackboard.com/phoenix.zhtml?c=177018&p=irol-newsArticle&ID=822607
HREF5 Blackboard Learning System - Vista Enterprise License - Reference Hardware Requirements
http://www.blackboard.com/clientcollateral/100506/Bb%20Learning%20System%20-%20Datasheet%20-%20Vista%20Hardware%20Requirements.pdf
HREF6 Moodle
http://www.moodle.org
HREF7 Sakai Project
http://www.sakaiproject.org/
HREF8 Open Source 2010: Reflections on 2007
http://connect.educause.edu/Library/EDUCAUSE+Review/OpenSource2010Reflections/40682
HREF9 Academic Hosting
http://www.moodlerooms.com/academic-hosting.html
HREF10 Supported Hosting
http://moodle.com.au/services.php
HREF11 Blackboard Academic Suite Hardware Sizing Guide
http://library.blackboard.com/docs/r6/6_1/admin/bbls_rel61_hardware_sizing_guide.pdf
HREF12 Web 2.0: Compact Definition?
http://radar.oreilly.com/archives/2005/10/web_20_compact_definition.html
HREF13 What is Web2.0: Design Patterns and Business Models for the Next Generation of Software
http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html
HREF14 What is Web 2.0? Ideas, technologies and implication for education
http://www.jisc.ac.uk/media/documents/techwatch/tsw0701b.pdf
HREF15 Mashing Up the Once and Future CMS
http://www.educause.edu/apps/er/erm07/erm0725.asp
HREF16 Library 2.0 Theory: Web 2.0 and Its Implications for Libraries
http://www.webology.ir/2006/v3n2/a25.html
HREF17 Web Mash-ups and Patchwork Prototyping
http://hdl.handle.net/2142/679
HREF18 Blackboard Features
http://www.utexas.edu/academic/blackboard/about/atut/surveys/facultyspring01/features.html
HREF19 The ECAR Study of Undergraduate Students and Information Technology, 2007 (Key Findings)
http://connect.educause.edu/library/abstract/TheECARStudyofUnderg/45075
HREF20 Pipes: Rewire the Web
http://pipes.yahoo.com/
HREF21 The Yahoo! User Interface Library (YUI)
http://developer.yahoo.com/yui/
HREF22 Windows Live Spaces
http://en.wikipedia.org/wiki/Windows_Live_Spaces
HREF23 MSN Spaces Fact Sheet
http://www.microsoft.com/presspass/newsroom/msn/factsheet/MSNSpaces.mspx
HREF24 Microsoft Popfly

http://www.popfly.ms
HREF25 Popfly Team Site
http://popflyteam.spaces.live.com/
HREF26 Microsoft Popfly: Yahoo Pipes for the rest of us
http://blogs.zdnet.com/microsoft/?p=452
HREF27 GoogleAlert #1: Google Launches World's Largest Search Engine
http://www.google.com/googlefriends/alert1_2000.html
HREF28 Amazon Elastic Compute Cloud (Amazon EC2) - *Limited Beta* FAQs
http://www.amazon.com/b/ref=sc_fe_c_0_201590011_5/103-0229499-8886213?&node=201591011
HREF29 Amazon Simple Storage Service (Amazon S3) FAQs
http://www.amazon.com/S3-FAQs-AWS-home-page/b/ref=sc_fe_c_0_16427261_7/103-0229499-8886213?node=16427271