

**Meta Course Tool for Blackboard Hosting
at the University of Cincinnati**

By

Karl Riddle

Submitted to the Faculty of the Information Technology Program
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Technology

University of Cincinnati
College of Engineering and Applied Science

June, 2010

**Meta Course Tool for Blackboard Hosting
at the University of Cincinnati**

by

Karl Riddle

Submitted to
the Faculty of the Information Technology Program
in Partial Fulfillment of the Requirements
for
the Degree of Bachelor of Science
in Information Technology

© Copyright 2010 Karl Riddle

The information in this document is proprietary and may not be reproduced or distributed
in whole or in part without the permission of the owner.

Karl Riddle

Date

Robert Schlemmer

Date

Hazem Said, Ph.D. Department Head

Date

Acknowledgements

The support of the UC Blackboard team was most invaluable in the completion of this project. Without the use of the department's test servers and the staff's support in developing the concept and testing the product this project would not have been possible.

I would also like to thank Professors Robert Schlemmer, Russell McMahon, Brandan Jones, Maria Damen, and my supervisor Frank Damen. Their constant support of my education over the past few years will never be forgotten.

Dedication

This project is dedicated to my lovely wife Meng and our soon to be born son Edward.

Table of Contents

Section	Page
Acknowledgements	i
Dedication	ii
Table of Contents	iii
List of Figures	iv
Abstract	v
1. Problem Statement	1
1.1 Definition of need	2
2. Description of the solution	3
2.1 User Profiles	3
2.1.1 Instructors	3
2.1.2 Deans/Department Heads	4
2.1.3 System Administrators	4
2.2 Design Protocols	4
3. Deliverables	6
4. Design and development	7
4.1 Timeline	7
4.2 Budget	7
4.3 Software and Hardware	8
4.3.1 Software	8
4.3.2 Hardware	10
4.4 Testing plan	10
4.5 Risk management plan	10
5. Proof of Design	12
6. Conclusion	13
References	14

List of Figures

Figure 1. GANTT Chart	7
Figure 2. Budget	8
Figure 3. Risk Matrix	11
Figure 4. Course Control Panel Link	12
Figure 5. MetaCourse Creation Wizard	12
Figure 6. MetaCourse Creation: Selecting Courses	13

Abstract

University of Cincinnati Blackboard Hosting hosts Blackboard for UC and thirteen other education institutions in the state of Ohio. Each school creates a Blackboard course for every official course. There are often multiple sections of the same course, and managing many Blackboard courses with redundant information is a burden to instructors and system administrators. While the Blackboard Learning System has many features, it lacks the ability to programmatically combine multiple courses and their respective enrollments. As a result, most institutions do not allow combined courses while others require a bureaucratic process which involves a system administrator manually combining courses. This project empowers instructors, department heads, and system administrators to easily combine many courses into one. The goal is to streamline the process and lessen the workload of support staff.

Meta Course Tool for Blackboard Hosting at the University of Cincinnati

1. Problem Statement

The Blackboard Learning System is the worldwide leader in the Learning Management System market. Educators use this system as an online repository for course documents, syllabi, and grades. It also allows students to submit assignments, take tests, and communicate with classmates through the Internet. Systems administrators can also take advantage of data integration tools which allow for the creation of courses and user accounts as well as the periodic updating of course enrollments. Although the system provides a wealth of functionality, most institutions find that it lacks some desired features. One feature that is sorely missing from Blackboard is the ability to combine courses.

UC Blackboard Hosting hosts the Blackboard Learning System for the University of Cincinnati and twelve other educational institutions in Ohio. The schools differ greatly with respect to the size of their student populations, but they all benefit from the experience of the UC Blackboard team and its ability to purchase hardware resources that would not be affordable to most schools. Another benefit is the customization that UC Blackboard Hosting offers through the development of Blackboard Building Blocks.

1.1 Definition of need

Combining courses, in Blackboard terms, means taking the student enrollments for two or more individual courses and placing them into a third course so that the instructor only has to post items and create tests once. There are many scenarios in which an instructor might want to combine courses: perhaps she teaches multiple sections of the same course, or she and a colleague want to share responsibilities and resources for the same or similar courses, or conceivably a class is dual-listed by the registrar so that students from two colleges can register for it separately. This project will create a tool that will empower teachers and systems administrators to combine courses.

Some schools do not even attempt to tackle this problem, while others have created elaborate solutions that require the involvement of various staff members and external systems. At the University of Cincinnati, if a faculty member wants to combine two or more sections of the same course, he has to follow a particular procedure. First, he must send an email to blackboard@uc.edu containing the course numbers to be combined. This email must be sent from an official UC email account, and the instructor must be considered “instructor of record” by the registrar, or else the request will not be processed. UC Blackboard staff members receive this email and then create an entry in a system external to Blackboard. That external system then creates enrollment files that have to be sent to the Blackboard server for processing. If the faculty member wishes to combine his course with a colleague’s course, both instructors need to send an email each and every time they wish to combine their courses. While this system serves its purpose,

it requires staff action and a separate Microsoft SQL database system, plus the instructor has to wait for a staff member to take action.

Blackboard Inc. does not include this functionality in its core product; however, the company offers a Building Block extension that can be licensed annually for a substantial cost. That cost is determined by the number of students that attend the college that wishes to license the product, but it is generally in the tens of thousands of dollars. Other vendors offer solutions which are cheaper than Blackboard's solution but still cost between \$5,000 and \$10,000 per year.

2. Description of the solution

The solution to this inability to easily combine courses is a Blackboard Building Block that is general enough to be installed on any Blackboard system running versions 7, 8, or 9 and which does not require the use of an external database or other software. It has configurable options such that systems administrators can manage permissions.

2.1 User Profiles

There are three categories of users: Instructors, deans and department heads, and system administrators.

2.1.1 Instructors:

- Merge two or more of his courses
- Merge one or more of his courses with those of another instructor
- Approve merging of a course with another instructor's course
- Add courses to a merged course
- Remove courses from a merged course

2.1.2 Deans/Department Heads:

- Merge two or more of his department's courses
- Approve merging of a course with another instructor's course
- Add courses to a merged course
- Remove courses from a merged course

2.1.3 System Administrators:

- Merge two or more courses
- Add courses to a merged course
- Remove courses from a merged course
- Update enrollments for merged courses

2.2 Design Protocols

Users enter this tool from one of two entry points: a course control panel link or a system administrator panel link. Only system administrators enter through the system administrator panel; department administrators and instructors enter through the course control panel. Upon clicking this link, users enter their meta course "home" screen in which they see meta courses they have previously created and meta courses which someone else created but which one or more of their courses belongs to, as well as a button to add a new meta course.

Clicking the button to add a new meta course takes the user through a wizard in which he or she provides a name for the meta course and then selects the children courses. The children courses can be courses taught by the user that is going through the wizard or by someone else. If the courses are taught by someone else, they will not be immediately added to the course. Instead, an email will be sent notifying the additional instructors that someone has requested their permission to add their course to a meta course. Instructors

who receive such a message must login to Blackboard and click on the meta course link in the course control panel to approve or deny the request to add their course to the meta course.

From the home screen, users also have the option to click a modify button to add additional child courses to a meta course they have created. Doing so takes them through part of the same wizard they used when creating the meta course, but this time it only allows them to add or remove courses.

The home screen is also where instructors who have been sent an email requesting authorization to add their course to a meta course will see that invitation. They will be able to click on either an approve or a deny button. Once one of those buttons has been clicked, they will only see that their course belongs to a meta course. To have the course removed after it has been approved, the user must contact the owner of the meta course, i.e. the instructor who created the meta course.

The user interface has been kept consistent with the various versions of Blackboard by utilizing the tag libraries that are available in the Blackboard API. These libraries provide special tags for formatting forms, buttons, and text, as well as other items. Blackboard 9 changes the interface of the course control panel, so this affected where the building block “hooks in” to the interface.

3. Deliverables

The deliverable is a Blackboard building block which can be installed on Blackboard versions 7, 8, and 9. This building block includes:

- Setup JSP which creates Oracle database tables
- Administrator interface that allows systems administrators to create and manage meta courses
- Faculty interface that allows faculty to create and manage meta courses
- Department Administrator interface that allows a department administrator to create and manage meta courses for many faculty members
- Notification and management feature that allows for automated messages to be sent when one faculty member wishes to join courses with another. The system should allow the recipient to click a link and log in to Blackboard in order to approve a request to merge their course with another faculty member's course.
- Enrollment updater process that periodically updates meta course enrollments to match those of the official courses that comprise the meta course
- Training materials that show faculty, department administrators, and systems administrators how to create and manage meta courses

The product meets the deliverable objectives; however at least one addition is planned for future development. Code should be developed to allow an instructor to export the grades for students in a meta course to a column in the official courses that belong to that meta course. The Blackboard Grade Center API is not documented so this will not be an easy task, but it is certainly worthwhile.

Figure 2

Expense	Projected	Actual
NetBeans	0	\$0
Java	0	0
SQL Developer	0 (included in Oracle license)	\$\$\$
Argo UML	0	0
Subversion	0	0
Labor	0	3,500
Blackboard Test Servers	0	\$\$\$
MacBook Pro	0	\$2,000

4.3 Software and Hardware

4.3.1 Software:

Java

UC Blackboard's application servers run the Red Hat Linux operating system. Blackboard can run on either Windows or Linux. Since .NET is out of the question without the Windows OS, this project will use Java Server Pages and Java classes (3).

NetBeans 6.7

While Java programs can be written with a tool as simple as Notepad, an Integrated Development Environment (IDE) provides features that ease the task of programming. NetBeans is the IDE of choice for many Java developers since it consistently provides support for new Java technologies (5). One of the best features is code auto-completion, which makes creating object based on an API much easier.

Oracle SQL Developer 1.5

This project requires the creation of multiple database tables in Oracle databases. Oracle SQL Developer is a package that is available with an Oracle database license and it will be used to facilitate the creation and testing of these tables (7).

Blackboard 7, 8, and 9 test servers

Unlike other websites or executable files, the only way to test a Blackboard building block is to install it on a Blackboard server. Since UC Blackboard hosting hosts 3 different versions of Blackboard, test need to be performed on each version. Therefore this project will utilize the existing Blackboard 7, 8, and 9 test servers hosted by UC Blackboard (9).

Argo UML 0.28

Before writing any code it is best to have a plan. Unified Modeling Language (UML) is the industry standard for defining classes and diagramming the interaction between them. There are many options to choose from but this project will utilize Argo UML since it is open-source (11).

Subversion 1.6.6

Software development is an iterative process. Sometimes new iterations produce desired results, and other times it is necessary to roll back to a previous version if things don't go well. Without a version control system, it would be necessary to manually create a backup copy of a project with every change. This is time consuming and naming standards would be absolutely critical.

Thankfully, programs like Subversion keep track of changes and make it easy to go back to a point in time to review old code or start over after making multiple changes. The program can also automatically increment version numbers in documentation (12).

4.3.2 Hardware:

MacBook Pro

The MacBook pro provided to me as an employee of the University of Cincinnati will be used to build this project. Since it will be written in Java, it can be written on any operating system.

4.4 Testing Plan

This project will be utilized by instructors and administrators at UC, as well as up to 12 other institutions on multiple versions of Blackboard. The testing plan involved installing the building block on a test server running each version of Blackboard. Users were created on those test servers, and the same tasks were performed on each: merging two courses taught by the same users, merging more than two courses taught by the same user, and merging one course taught by one user with a course taught by another user. Once courses were merged, the tester also attempted to add additional courses or remove existing child courses from the meta courses. Each of these tasks will be performed by different user accounts with the roles of instructor, department admin, and system administrator.

4.5 Risk Management Plan

Figure 3 outlines the potential risks that may hinder the development process and the mitigation for each. The project will be completed on time, but already some setbacks have presented themselves. Since the Blackboard Hosting Project has been inundated with projects in the past few months, as well as some unexpected Oracle data corruption, it has been difficult to find sufficient time to dedicate to this project. Nevertheless, much time has been spent on it and it will be completed on time.

The changes that were made to the Blackboard API and tag libraries in version 9 are drastic, but they should not hinder development. At worst, the interface on the building block will more closely resemble previous versions if the new tag libraries prove too difficult to use. This will not cause functional issues, however, since the classes used in this project remain mostly unchanged in version 9.

Figure 3

Risk	Level of Risk	Mitigation
Potential problems with getting code to work in Blackboard 9 since the API and user interface has many changes (Technology Risk).	Medium	Each individual component of the project will be developed simultaneously on each version of Blackboard. Since ideally the code would be identical on all versions, this should minimize the number of times code has to be re-written (Risk mitigation).
Blackboard may decide to include this functionality in its product for free (Market Risk).	Low	Blackboard and other companies are currently making a good deal of profit on this sort of tool, so it is in its best interest not to discontinue this line of business (Risk Acceptance).
Differences in expectations of functionality due to the various groups of users at the 12 schools that UC Blackboard hosts (Process Risk).	Medium	Since this product will be used by a variety of people at a number of distinct institutions, input will be gathered from as many of the involved parties as possible to make sure as many configuration scenarios as possible are taken into account (Risk Mitigation).
Unfamiliarity with Subversion software poses a challenge (Technology Risk).	Low	Subversion is not a requirement for the functionality of this project; it is simply a way to keep track of changes. If Subversion should prove to be too difficult to use or too much of a burden it can be excluded from the project without harming progress (Risk Acceptance).
If workload becomes too high at work, it may be difficult to spend time on this project.	High	My supervisor is very supportive of this project and. Hopefully he will allow me to work on it (Risk Acceptance).

Blackboard Inc. has not made any mention of offering this functionality for free; it continues to charge a hefty license fee for similar functionality, so there is little chance that a course merge tool will be introduced to the core product.

5. Proof of Design

Figures 4, 5, and 6 below show screen captures of some of the steps required to create a meta course. Figure 4 shows the page where an instructor must click on the Meta course link to get to the home page.

Figure 4

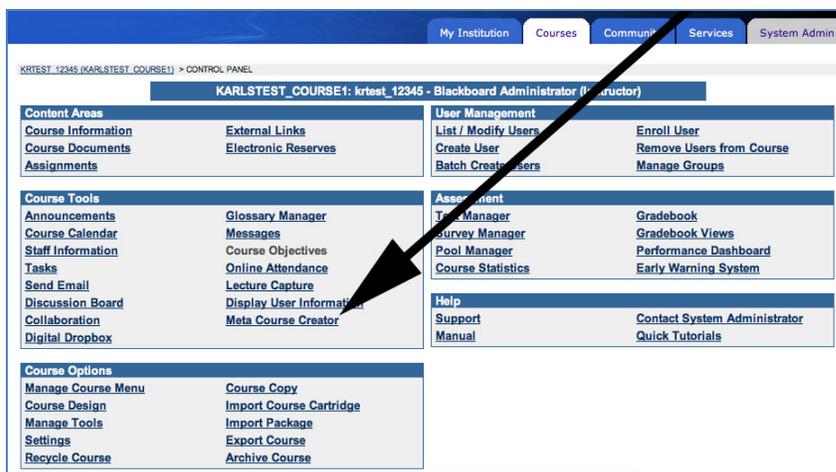


Figure 5 shows the page where a user provides a title for the meta course.

Figure 5



Figure 6 shows the screen where a user selects the courses which will be included in the meta course.

Figure 6

Create Meta Course: English 101

1 Course List

Select the courses that you wish to merge.

Course Title	CourseID	Instructor
<input type="checkbox"/> Exam View Test Elder	eld-examview	
<input type="checkbox"/> cass	casstest1	
<input type="checkbox"/> karl_test0306_9	karl_test0306_9	
<input type="checkbox"/> karl_test0307_1	karl_test0307_1	
<input type="checkbox"/> krtest_12345	karstest_course1	
<input type="checkbox"/> null	null	
<input type="checkbox"/> test_karl0307_3	test_karl0307_3	

2 Submit

Click **Submit** to finish. Click **Cancel** to quit.

6. Conclusion

This Meta Course Tool provides new functionality for all clients of UC Blackboard Hosting, and it enhances the functionality that is currently offered to UC's Blackboard users. Instructors will be empowered to create their own meta courses, and system administrators will no longer be required to be involved in every step of the process of creating a meta course. The project will be installed on clients' Blackboard systems during the summer of 2010. It will also be presented at the Blackboard Developers Conference in Orlando, Florida on July 13, 2010.

References

- 1) **Blackboard Extensions** [Online] / auth. Blackboard Inc. // Blackboard Course Merge Tool. - November 26, 2009. - <http://www.blackboard.com/Support/Extensions/Extension-Details.aspx?ExtensionID=23014>.

- 2) **Enhancing E-Learning with Media-Rich Content and Interactions** [Book] / auth. Caladine Richard. - Hershey, PA : Idea Group Inc, 2008.

- 3) **Library 2.0 initiatives in academic libraries** [Book] / auth. Cohen Laura B.. - Chicago, IL : Assoc of College & Resrch Libraries, 2007.

- 4) **Merge Course Enrollments** [Online] / auth. Global Synergies // Global Synergies. - December 1, 2009. - <http://globalsynergies.com/merge/>.

- 5) **NetBeans IDE field guide: developing desktop, Web, enterprise, and mobile applications** [Book] / auth. Patrick Keegan Ludovic Champenois, Gregory Crawley, Charlie Hunt, Christopher Webster. - [s.l.] : Prentice Hall, 2006.

- 6) **oln.org** [Online] / auth. University System of Ohio // University System of Ohio. - December 2, 2009. - http://www.oln.org/emerging_technologies/cle_hosting.php.

- 7) **Oracle Essentials, 4th Edition** [Book] / auth. Greenwald Rick, Stackowiak Robert and Stern Jonathan. - Sebastopol, CA : O'Reilly Media, Inc., 2007.

- 8) **Overview of Building Blocks and PowerLinks** [Online] / auth. Kroner George // edugarage.com. - 1 26, 2009. - December 1, 2009. - <http://www.edugarage.com/display/BBDN/Overview+of+Building+Blocks+and+PowerLinks>.

- 9) **Release 9 How To Upgrade** [Online] / auth. Blackboard Inc. // blackboard.com. - December 1, 2009. - <http://www.blackboard.com/Release9/Vista-8/Steps-1-2-3.aspx>.

- 10) **UC Blackboard Help** [Online] / auth. University of Cincinnati. - December 4, 2009. - https://bbservices.uc.edu/documents/help_site/knowledgebase/display.asp?id=145.

- 11) **UML 2 for dummies** [Book] / auth. Michael Jesse Chonoles James A. Schardt. - New York, NY : For Dummies, 2003.

- 12) **Version Control with Subversion, 2nd Edition** [Book] / auth. Pilato C. Michael, Collins-Sussman Ben and Fitzpatrick Brian W.. - [s.l.] : O'Reilly Media, Inc., 2008.