AC 2007-346: DOCUMENT MANAGEMENT IN TEAM-ORIENTED, PROJECT-BASED COURSES: EVALUATING A LATEX/SUBVERSION-BASED APPROACH

Sandra Yost, University of Detroit Mercy

Mohan Krishnan, University of Detroit Mercy

Document Management in Team-Oriented, Project-Based Courses: Evaluating a LaTeX/Subversion-Based Approach

1 Abstract

This paper discusses a low-cost approach to the implementation of a document versioning system for technical reports. Several alternatives have been considered, including commercial document collaboration services such as NextPage 2™ (NextPage, Inc.) and SharePoint™ (Microsoft Inc.), open-source versioning applications such as Subversion or CVS, wikis, and free web-based services such as Google Docs & Spreadsheets (formerly Writely). This paper explores these alternatives and then focuses on a versioning system-based solution as the approach judged most appropriate for our requirements.

2 Introduction and Background

Most engineering and technology programs place a high value on team-based assignments and projects. At the University of Detroit Mercy, a written project report is often one of the required deliverables from each team. When the size of the team exceeds two or three, collaborative report writing becomes problematic. Ad hoc processes aimed to keep track of who is working on what, and which version is actually the correct current version often break down, and chaos ensues. It is also difficult for an instructor to find clear evidence of an individual's contribution to the report.

The typical document revision cycle for a student team goes something like this:

1. Student A creates a draft outline of the document in Word and includes a draft of the section for which he/she is responsible.
2. Student A e-mails the document to teammates, usually with some indication as to whose turn it is to revise the document (Student B). Problems arise if more than one person tries to revise the original document. In such cases, merging the changes from several revised documents is not always straightforward.
3. After adding other content and/or revising Student A's content, Student B e-mails the document to the next team member.
4. Eventually, the document comes back to Student A, and another cycle of revision can begin.

Students often wait until the last minute to begin project reports, and this ad hoc system often breaks down. A team member may go missing just when teammates are waiting for an e-mail with revisions from her/him. It often falls to the team leader (or Student A in the above scenario) to plug as many gaps as possible at the last minute, based on what may be a limited knowledge of the details of the part of the project in question. This often results in project reports that read as though they were slapped together at the last minute (because they *were*).

Changing the revision cycle so that all team members can work simultaneously on the document may not prevent the last-minute scramble to write the report, but it can address the delays that are introduced in waiting for the document to travel sequentially among the team members.

This issue has been addressed in several publications. Miertschin and Willis[1] discuss the use of blogs and Microsoft SharePoint™for electronic collaboration among Information Systems Technology students. Buchal[2] also addresses the use of SharePoint™. Raman et al.[3] discuss the use of a wiki for collaboration in a graduate-level knowledge management course.

An informal survey of 3rd year Electrical Engineering students at the University of Detroit Mercy revealed that they do not even use the "Track Changes" built-in feature of Microsoft Word™ when collaborating on a document. One might ask why we didn't first instruct the students in using this feature before looking for a more complex solution.

There are two significant drawbacks to simply using the track changes feature for team document collaboration. First, once changes to a document are accepted by the original author, the history of changes to the document is no longer available unless the revised document is saved with a different file name. Doing so creates multiple versions of the document, also creating more confusion when different team members are working on different versions of the document. The workaround to this first issue might be to delay accepting any changes until all cycles of review and revision have been completed. However, the markup created by Word can become quite convoluted, as authors make revisions on top of earlier revisions. See Figure 1 for an example of the markup created by Word.

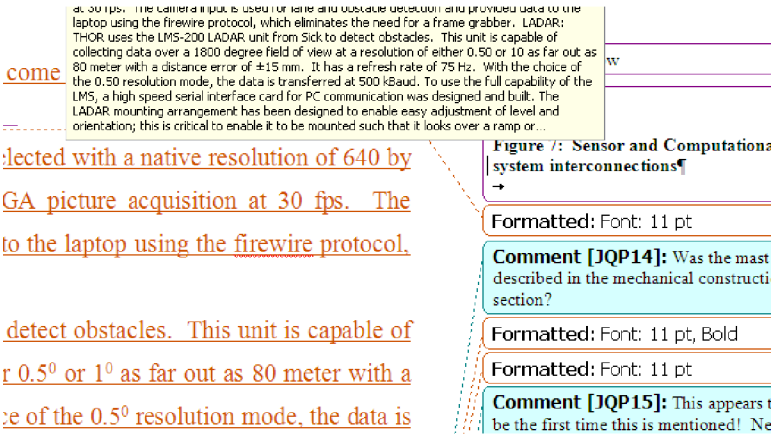

Figure 1: Screenshot of Word Markup

In considering how to achieve simultaneous revision by all team members, it was critical to determine the desired features of a collaborative authoring system. We specified the following features as necessary to achieve our goals:

- The document must be available to all team members at all times.
- Changes submitted by one team member cannot be lost when another team member submits changes to the same revision.
- The system must maintain a history of changes to the document so that one could revert to an earlier version if necessary.
- The system must be able to track changes to equations as well as to other text.

3 Some options for document collaboration.

In this section we identify and discuss some tools for team collaboration on technical documents. Note that the options discussed here are a subset of the possible tools available.

3.1 Commercial document collaboration solutions.

Both Microsoft's SharePoint™ and NextPage's NextPage 2™ are document management systems that incorporate versioning.[4,5] They also include sophisticated features such as the ability to assign different levels of permission to the various members of a collaborative team; for instance, the right to create a new version. They can also send e-mail notification to the team members upon the creation of a new version.

Because they are commercial packages, they tend to be costly; the cost is reflective of the fact that they are designed to be relatively bulletproof and forgiving in the face of assaults from the uninitiated user. Additionally, they incorporate a number of advanced features that may never or only rarely be exercised by the average user. However, if a university already possesses a site license for one of these products, it could be a viable option. It is unknown how these commercial packages treat equations. Because of budget constraints, such systems were not seriously considered.

3.2 Google™ Docs & Spreadsheets (formerly Writely).

Writely was investigated as a collaborative writing tool alternative before the project described in this paper was first initiated in August 2006. It was a web-based word processor which incorporated some basic features of document management such as monitoring and reversing changes and was offered free of charge. However, it did not support the inclusion of mathematical equations - these were saved as graphics objects when the document was uploaded to the server. This made it impossible to make minor changes to an equation. Since this is an important aspect of engineering reports, Writely was not chosen. Subsequently Writely has been acquired by Google™ and is now offered as part of Google™ Docs & Spreadsheets.[6] It is still free of charge. Since the product is now backed by a company with considerable resources, it is worthwhile to keep an eye on its evolution.

3.3 Wikis as a tool for team document production.

Wikis maintain and provide a history of edits made to articles. This allows multiple authors to contribute to the wiki without danger of losing previous versions. This feature would be useful in the event that a team member inadvertently deletes part of the document or makes a change with which the rest of the team disagrees. In such a case, the previous version can be retrieved.

The MediaWiki[7] engine supports an add-on for equations, an important consideration for the preparation of technical reports. It is open-source software that works well on an Apache server. We had the IT team at the university install it on a test server for evaluation. It didn't take very long to conclude that a wiki was not the ideal solution to the problem. The drawbacks to this approach are:

- A wiki is in essence a collection of discrete web pages (articles). The organization of these

pages depends on the links inserted by the authors. To create a technical report, it is critical that the pages be linked in sequential order. We believe that searching out and correcting erroneous and missing links would distract students from the content of the report. We also felt that while a wiki might be appropriate for presenting the report as an online document, it does not allow one to easily produce a professional-looking printed document.

- Wikis are generally designed to allow anyone to view, and in many cases, edit the content. In order to set up MediaWiki to restrict access to only certain individuals, a somewhat complicated workaround had to be implemented by our IT team. It occurred to us that using the wiki in this way was somewhat like trying to turn a lion into a herbivore.
- While MediaWiki supports equations, authors must know some LaTeX-formatting to insert equations.

3.4 An open-source versioning system solution.

Versioning systems (e.g. CVS[8], Subversion[9]) have enjoyed extensive use by the open-source software development community because they allow many users to work on a project while keeping track of the complete history of changes and the most current version of the code. Unfortunately, these systems operate on source code stored as text files and are not designed to work with Word or other documents stored as binary files. However, they do work with LaTeX files, which are stored as text files. The important issue here is whether or not compelling students to learn LaTeX places an unnecessary burden on them. Because the only other viable option (MediaWiki) requires some use of LaTeX coding anyway (for equations), we felt that getting students to use LaTeX to format the entire document would not be an unreasonable burden, provided that we supply templates that can be easily adapted by the students for specific assignments.

Using a versioning system also provides us as instructors the ability to query the repository to see what contributions were made by individual students at various stages of the document creation process.

4 Pilot Study Design

We have just conducted an initial pilot study to evaluate whether or not and how a LaTeX/Subversion-based approach might be adopted by the EE program as a standard for the preparation of team-based project reports. A follow-up study is underway for the Winter 2007 semester.

4.1 Using LaTeX to produce technical documents.

LaTeX[10] is a set of macros based on the TeX[11] typesetting language. It is widely used in the preparation of manuscripts in mathematics, science and engineering. Many engineering graduate schools encourage degree candidates to use LaTeX to prepare theses and dissertations. The engineering education literature contains few direct references to the use of LaTeX in undergraduate programs[12,13]. These references pertain to the use of LaTeX in physics and mathematics undergraduate curricula.

Students in the Fall 2006 Electronics II course received instruction in document preparation using LaTeX during the second week of October. A circuit simulation assignment which required them

to use LaTeX for the report followed. (Note that the introduction of the versioning system took place later in the semester in order to give the students some time to learn LaTeX basics first.)

To ease the learning curve for students, we created a sample document template for the students to use. This freed them from much of the formatting burden, allowing them to focus on the content. It also ensured that we would get the final documents from students in the desired format.

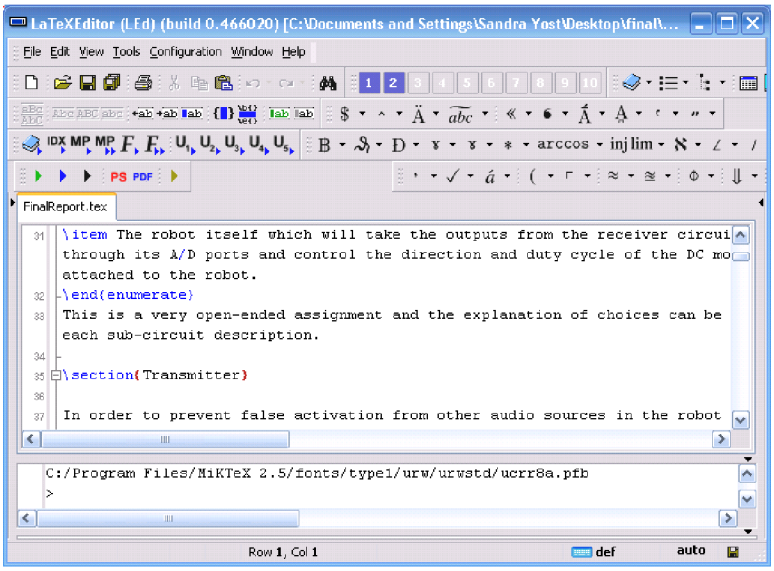

Figure 2: Screenshot of LaTeX Editor

Because our students are used to the Windows environment, they are not generally comfortable with launching applications from a command-line tool. We found a free application called LaTeX Editor[14] that features context-sensitive editing of LaTeX documents, and provides launch buttons to do the LaTeX and BibTeX compilations. Figure 2 contains a screenshot of this editor.

4.2 Using Subversion to facilitate team collaboration on documents.

We introduced Subversion during the second week of November 2006 after students had had an opportunity to create a project report using LaTeX. With the help of the IT department, a Subversion repository was set up on an Apache server. We created an account for each student for accessing the repository. Nine students worked as a team to design, build, test, and integrate different subsystems of a complicated electronic system. The system needed to use unique combinations of different tone signals to enable the remote control of a mobile robot. A single comprehensive project report was required at the end of the semester with contributions from the various team members.

Students interacted with the repository using the free foundation version of an application called SmartSVN.[15] They learned how to check out files from the repository and commit local changes back to the server. Figure 3 shows a sample screenshot from the SmartSVN client showing that some files on the local machine have been modified from the latest version available on the server.

Subversion also allows a side by side comparison of different versions of a document. Figure 4 shows an example of such a comparison. The user may update his or her working copy of the

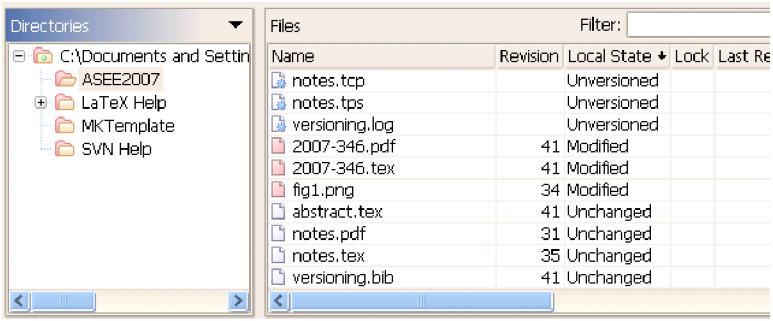

Figure 3: SmartSVN status screen

document with the latest version found in the repository, or accept only selected changes. Once the working copy is updated and further edited, the user must commit these changes to the repository so the latest revisions are available to team members.

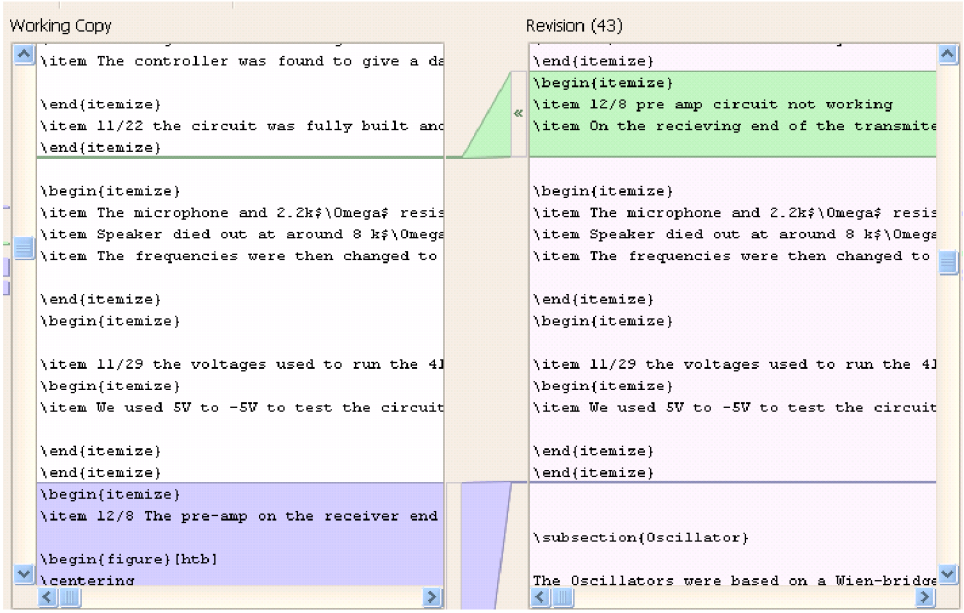

Figure 4: SmartSVN compare screen

5 Discussion and Next Steps

After the introduction to LaTeX, students completed an in-class activity that required them to create a LaTeX document having many features, including equations, bibliographic citations, figures, and tables. After completing the activity, students were asked to complete a survey regarding their confidence in using LaTeX. The survey used a five-point Likert scale. The results from the 8 students (out of 9 enrolled in the course) who completed the survey are shown in Table 1.

At the end of the semester, two items were added to the final course survey to assess student confidence with both LaTeX and Subversion. Table 2 contains these results.

The early results in Table 1 seem to indicate a reasonably high level of confidence in using LaTeX,

Survey Item	SD	D	N	A	SA
I feel comfortable with compiling LaTeX documents from the LEd editor.		1	1	6	
I know how to view the PDF output after I compile a LaTeX document.			1	2	5
If I rename the template files, I know where to change references to these files in the main .tex document.			4	3	1
I know how to insert equations into my document.		1	2	4	1
I know how to insert graphics (figures) into my document.			2	5	1
I know how to insert tables into my document.			2	6	
I know how to add references to the .bib file and cite them in the .tex file.		1	1	6	
If the versioning system (Subversion) were introduced now so that we could use it with our first project, I would feel too overwhelmed learning both LaTeX and Subversion at the same time.			4	3	1
My team would like to start using Subversion as soon as possible.		1	7		

Table 1: Initial Survey Results

or at least in the individual formatting tasks identified in the survey. However, the responses to the last two items seem to indicate at least some ambivalence to LaTeX in that students did not seem eager to move right into the versioning system without more practice in LaTeX.

The final survey (Table 2) asked a more comprehensive question about LaTeX instead of breaking up questions about the use of LaTeX into questions about individual formatting tasks, so it is difficult to compare the early and later results. It might seem that students had less confidence at the end of the term than at the beginning, but it is impossible to know this with any level of certainty because of the fundamental difference in the way the questions were asked. It will be important to match the pre- and post- surveys more carefully in future studies.

Survey Item	SD	D	N	A	SA
I can use LaTeX to produce a technical document.	0	0	6	2	1
I know how to collaborate on a technical document using a versioning system (Subversion)	0	1	6	1	1

Table 2: Final Survey Results

Several students submitted written comments to let us know that they thought this LaTeX/Subversion approach to team document preparation was something they thought would be valuable in team settings, especially in the capstone design course sequence in the senior year.

5.1 Follow-up to initial pilot study

During the initial pilot study, we kept a detailed log of our observations of students' use of LaTeX, and then Subversion. Some unanticipated difficulties some of the students encountered helped us in improving the instruction for the next cohort of students.

Some of these problems included the following:

- Not extracting the template files from the ZIP archive before trying to compile.
- Trying to compile one of the secondary `.tex` files or the `.bib` file instead of the primary document.
- Using the same reference label for multiple figures or equations.

Two levels of follow-up are planned. The students who just completed the Electronics II course are currently on co-op rotation and will return to take courses in the second Junior semester in May 2007. Another co-op rotation will take these students off campus in September 2007, and they return in January 2008 for two consecutive academic semesters that include the capstone design course sequence. It will be important to work with instructors in the Summer 2007 term to encourage the continued use of LaTeX/Subversion so that the students will still remember it for the capstone design course sequence.

The second level of follow-up is to introduce LaTeX/Subversion at an earlier stage in the students' academic program. In January 2007, the first author began teaching three second-semester sophomore courses, and is in a position to work with these students to produce better team project reports using LaTeX/Subversion. Lessons learned from the first pilot study should lessen some of the difficulties experienced by the Juniors in learning how to participate in this collaborative writing effort.

6 Acknowledgments

The authors would like to thank Mr. Kevin Hickey, University of Detroit Mercy Web Services Team for his technical support of this project.

References

[1] S. Miertschin and C. Willis, "Building infrastructure to develop electronic collaboration skill among students," in *Proc. 2005 ASEE Annual Conference and Exposition: The Changing Landscape of Engineering and Technology Education in a Global World, Jun 12-15 2005*. Portland, OR, United States: American Society for Engineering Education, Chantilly, VA 20153, United States, 2005, pp. 1357–1367.

[2] R. O. Buchal, "Web-based shared workspaces for collaborative learning," in *Proc. 2002 ASEE Annual Conference and Exposition: Vive L'ingenieur, Jun 16-19 2002*. Montreal, Que., Canada: American Society for Engineering Education, Washington, DC 20036, United States, 2002, pp. 6985–7001.

[3] M. Raman, T. Ryan, and L. Olfman, "Designing knowledge management systems for teaching and learning with wiki technology," *Journal of Information Systems Education*, vol. 16, no. 3, pp. 311–320, Fall 2005.

[4] Microsoft SharePoint, Retrieved January 3, 2007, http://www.microsoft.com/sharepoint/.

[5] NextPage 2, Retrieved January 3, 2007, http://www.nextpage.com/products/collaboration/index.htm.

[6] Google Docs & Spreadsheets, Retrieved January 3, 2007, http://docs.google.com/.

[7] MediaWiki, Retrieved January 3, 2007, http://www.mediawiki.org/wiki/MediaWiki.

[8] Concurrent Versioning System (CVS), Retrieved January 3, 2007, http://www.nongnu.org/cvs/.

[9] Subversion, Retrieved January 3, 2007, http://subversion.tigris.org/.

[10] L. Lamport, *A Document Preparation System: LaTeX, User's Guide and Reference Manual*, 2nd ed. Addison-Wesley, 1994.

[11] D. Knuth, *The TeXbook*. American Mathematical Society and Addison-Wesley, 1984.

[12] J. R. Taylor and B. A. King III, "Using computational methods to reinvigorate an undergraduate physics curriculum," *Computing in Science and Engineering*, vol. 8, no. 5, pp. 38–43, 2006. [Online]. Available: http://dx.doi.org/10.1109/MCSE.2006.100

[13] J. J. Westman, "Introduction to scientific research: Research experiences for undergraduates," in *Proc. 2002 American Control Conference, May 8-10 2002*, vol. 2. Anchorage, AK, United States: Institute of Electrical and Electronics Engineers Inc, 2002, pp. 1092–1096. [Online]. Available: http://dx.doi.org/10.1109/ACC.2002.1023164

[14] LaTeX Editor, Retrieved January 3, 2007, http://www.latexeditor.org/.

[15] Syntevo Software Development, "SmartSVN," Retrieved January 3, 2007, http://www.syntevo.com/smartsvn/.