

Adapting Lightweight Source Control and Project Management Software for Use by Multidisciplinary Product Design Teams

R. Keith Stanfill and Ethan I. Blackwelder
University of Florida

This paper will share our experiences with adapting an open-source, Source Control Management (SCM) and project management tool for use by multidisciplinary product development teams and remote sponsor clients.

Students working together to create design project artifacts such as reports, presentations, digital models, and software quickly learn that relying on their personal computers and e-mail accounts for collaboration and storage of the most up-to-date content leads to disaster. One can never be sure that they are working on the latest version of the digital file. Project failure is one hard disk crash, e-mail inbox overflow, or home network crash away. Further, external clients (project sponsors), may be uncomfortable with company-sensitive data residing on student personal computers. Providing on-campus laboratories can help, but requiring students to come in to the lab to do project work reduces the hours that students can contribute—especially if their most productive hours are outside of the lab hours. Restricting project data to in-lab access also precludes external client collaboration.

To satisfy the needs of the project teams and the project sponsors, an open-source web-based system was customized to provide a wiki, project tracking tools, and integration with a Source Control Management (SCM) tool. Participating students, faculty, staff, and sponsor clients all have authorization-based network access to team data. SCM tools have been found to be effective for managing any documents produced by the teams and provide recovery from document handling mishaps such as deletions and lost drafts. The tool helps the teams impose a structured approach to developing collaborative documents.

Introduction

In 1995 the concept of an academic product development team employing a web page to collaborate was a novel concept. At that time social networking tools, such as chat rooms, were restricted to service providers such as CompuServe, Prodigy, and America Online. Instant messaging debuted in 1996, and in the time since, social networking web sites, blogs, threaded discussion groups and text messaging have since seen explosive growth. The so-called “Millennials” were raised on these technologies and rely on them daily. Most recently, companies such as Parametric Technology Corporation are developing tools for “Social Product Development.”

Our multidisciplinary product development program, Integrated Product and Process Design (IPPD), began in 1995¹⁻² along with our attempts to provide collaboration tools for our project teams. Data access evolved from a mixture of data storage on in-lab computers and e-mail downloads, where sponsor clients had no network-based access to team data, to today’s wiki- and source-control-based system. The web-based system provides a wiki, project tracking tools, and integration with a Source Control Management (SCM) tool. Participating students, faculty, staff, and sponsor clients all have authorization-based network access to team data.

While the SCM tool is ideal for software development projects, we have found that it is effective for managing any documents that need to be versioned. Students and clients can collaborate on documents and merge changes with assistance from the SCM tools. Versioning tools also allow the team to roll back to previous document revisions. We plan to use our experiences with this open source collaboration approach to guide our search for commercially supported tools. Eventually we plan to adopt integrated tools that will fully enable “Social Product Development.”

Motivation

Starting in 1998, IPPD began using a server to manage team project data and provided web space for teams to post project content on a webserver. Access to the web content required accounts to be created for the sponsor company’s liaison engineers. Teams appointed a web master to post project deliverables and keep the website current. Web content could only be added by the team’s webmaster, and only when the web master was physically in the lab and logged onto the server. Web masters quickly became bottlenecks as they could not keep up with the posting of project content. Posting of project content on public web sites, especially private

company data, was not allowed, although occasionally a project team would be caught doing so.

Over the years, a groupware tool, Groupwise, was utilized with little success. License costs for Groupwise and the associated Novell server products eventually became unattractive financially as more units in the university moved to other server operating systems.

The authors first learned about using wikis for collaboration while networking at various National Collegiate Inventors and Innovators Alliance (NCIIA) and American Society for Engineering Education (ASEE) conferences. A wiki is a web portal that allows groups to contribute and edit web content from within a web browser, without the need to have special web page editing software. The wiki could therefore eliminate the single webmaster bottleneck and provide a more collaborative web experience.

The following key wiki features were deemed necessary for the IPPD program:

1. Role-based authentication using technologies already in place on campus
2. Hosting on IPPD-owned servers
3. Secure environment
4. Customization with common web development tools (such as PHP, javascript, python, PERL, etc.)
5. Project management capabilities
6. Integration with Source Control Management (SCM) or file versioning tools
7. Ability to upload attachments
8. Simple wiki formatting language
9. Ease of implementation and maintenance
10. Page history
11. PDF output

A search of wikimatrix.org³ indicated that there are at least 123 wikis available. A web application developer working for IPPD culled the list of wikis down to a few and from that list, a pilot implementation of MediaWiki⁴ was launched. Shortcomings with MediaWiki, such as lack of SCM, led to the deployment of the Trac wiki/SCM system. SharePoint from Microsoft was considered, but the perceived expense of supporting server hardware and end user licensing precluded serious consideration.

Adaptation

We face multiple issues when adapting these tools for our projects. Each year, our program has 23 to 27 teams of approximately 5 to 6 students, each mentored by faculty coaches and liaisons engineers. Configuring user- and group-based authentication schemes for approximately 150 users every program year has its challenges. Also, when working with IPPD corporate sponsors, project teams may work on copyrighted,

trademarked, or patented materials; thus overall security is critical—even between teams. Simply installing and configuring any software package 25 times—while providing secure web-access—is also difficult. Other issues include the ability to easily migrate, replicate, and archive project data.

In the 2007-2008 IPPD program year, student and faculty teams were given installations of MediaWiki—a popular, open source wiki engine written in PHP—to manage their project documentation. The chosen implementation method made it extremely difficult to migrate the installations to a different server for backup or for distributing the team’s data to project sponsors.

Upon receiving favorable reviews from colleagues teaching software engineering courses, we investigated Trac⁵, an open-source, “enhanced” wiki engine and issue-tracking system for the 2008-2009 IPPD program year. Core features of Trac include a wiki engine, an issue-tracker, a source code browser, roadmaps and milestones (for project planning), an integrated timeline, and search features. There is a web site dedicated to community-supported Trac plug-ins⁶ that augments the minimalist “out-of-the-box” wiki feature-set to be largely on par with MediaWiki (for example, we incorporated a WYSIWYG editor plug-in for editing wiki pages and an approximation of MediaWiki’s wiki markup processor).

For the Trac installations of 2008-2009, we developed a custom authentication plug-in for students and faculty that made use of a university-wide cookie-based authentication system; this was used to authenticate students and coaches. We also re-used our LDAP server from the previous year to authenticate remote liaison engineers who did not have university user accounts. This was supported by Trac out-of-the-box. We also extended Trac’s built-in name- and group-based authorization plug-in to consolidate user-authorization for all projects into a single data source (in this case a flat-file).

In 2009, the university deployed Shibboleth⁷—a secure, open source, single-sign-in system for web applications—to replace the (less secure) cookie-based system of the previous year. They also provided us with a locally managed branch of the university’s Active Directory system whose attributes feed into the Shibboleth system. To simplify management, the local LDAP server was abandoned. This change forced remote liaison engineers to obtain user accounts from the university (as the students and faculty already do). Usernames and passwords for all IPPD users are now managed by the university’s systems. We then developed a custom Trac plug-in that read user authentication attributes from Shibboleth and reverted to use Trac’s built-in authorization plug-in. Because Shibboleth provides a single sign-on system, no extra sign-on steps are required when moving between

various IPPD web-applications, even though they may be on separate servers or written in different languages.

Although not every project team deals with software, each project team deals with documents and data that are versioned. Additionally, all teams need a “single source of truth,” accessible securely while remote from campus. To address versioning and remote access needs, each team was provided with its own version-control system (VCS) to better manage their design documents. Subversion⁸ is a reliable VCS with excellent client programs on all major operating systems. Large numbers of Subversion repositories can be easily installed via scripting; they are also easy to back-up, replicate, and migrate. The 2008-2009 IPPD Subversion repositories were deployed to use the IPPD-managed LDAP system for user authentication and group-based authorization. Since the students and faculty were still using university-provided accounts to access their Trac resources, they had to use multiple passwords and/or usernames when going between applications, often leading to confusion amongst the users.

In 2009-2010, the IPPD-managed LDAP server was discarded in favor of Shibboleth to authenticate users for Trac. Subversion will not work with Shibboleth, but an LDAP view of the university's Active Directory was available—the same branch that we manage for Shibboleth. This change resulted in less account management for IPPD staff members.

Deployment

Although there are some repeat coaches and liaisons, we do have new coaches to train each new program year. Our students are more than happy to graduate, so we have no repeat students in successive years. As a result, between August 2008 and January 2010, we have overseen Trac and Subversion training for more than 250 people. We found deployment of the software to our IPPD users to be a non-trivial task.

In the first year deploying Subversion, we had to train both ourselves and our students and coaches in how to use Subversion. At the beginning of the program year, we provided each team with access to their own Subversion repository. They were told that they could use these repositories for storing project documentation, meeting notes and agendas, trial data, source code, and other shared files. Our website developer spent a class period at the beginning of the year demonstrating version-control basics such as how to upload and download files, and merge changes. We uploaded help topics to the IPPD course wiki and attached screen-capture videos, power-point slides, and links to outside resources. In our two computer labs, we downloaded each repository onto the team's share-drive to help jump-start the process. At the conclusion of the first program year, we found to our dismay that at least some

of the students who successfully used the lab computers to access their repository were un-aware that the repository could also be accessed from home. We also found that some students failed to understand fundamental version-control concepts (for example, the ability to “update a working copy”) and were wasting time and energy as a result.

For the 2009-2010 IPPD program year, a train-the-trainer model was adopted. Each team elected a “wiki-master” to become an expert in Trac and Subversion and in turn train their team-members, coach, and liaison engineers. Early in the Fall 2009 term, the IPPD web developer gave a training lecture to just these students. The help topics and share-drive accounts were also set up the same as the previous year (and were still accessible to all IPPD users).

In tandem with Subversion training at the beginning of each program year, we provided an overview of Trac's core features: wiki engine, source code browser, issue-tracker, and project planner. We also provided demo Trac installations showing what a successful roadmap (milestones with corresponding tickets) may look like. We also fielded the occasional request to install additional plugins: for example a plugin to display math equations written in LaTeX format.

Comparatively, we found that end-user training is significantly easier for Trac—in large part because the wiki paradigm is a significantly easier concept to grasp. Before you can even begin demonstrating how Subversion works, you have to define essential concepts on an abstract level: repository vs. working copy, commit vs. checkout vs. update, and so on. We also recognize the growing popularity of wikis—Wikipedia being a prime example.

Although we have to train many of the students in both of these tools each year, there are a few benefits of doing so: we and the returning coaches are better able to aid in the training of the new students. We are also accumulating more training resources.

Assessment

At the midpoint of the 2009-2010 program year only 16.67% of the teams failed to use Subversion, compared to 39.13% from the previous year midpoint (see Table 1). Year two usage data shows a 65% increase in the average number of revisions per team. The increase in usage may be attributed to the train-the-trainer approach and the returning faculty coaches who are themselves more experienced with Subversion.

The 2008-2009 IPPD program year project teams averaged about 26 wiki page edits and 9 attachments a month. The 2009-2010 teams have 33% less wiki edits and 50% fewer attachments (see Table 2). The decrease in the number of attachments and wiki edits indicates a better usage of Subversion as a document-sharing tool.

It is also possible that some of the 2009-2010 teams are using Google Docs.

Table 1 *Comparison of Subversion usage by IPPD program year*

Program year	# Teams	Total revisions	# Revisions/team	% Teams w/o revisions
2008-09*	23	1404	61.04	39.13%
2009-10**	24	4190	174.58	16.67%
% Change			+65.03%	-57.41%

Note. *As of January 20th, 2009. **As of January 20th, 2010.

Table 2 *Comparison of Trac usage by program year*

Program year	Average # of attachments	Average # of tickets	Average # of milestones	Average # of wiki edits	Average total Trac usage
2008-09*	45.22	5.39	8.74	128.70	188.04
2009-10**	21.63	3.83	2.54	85.04	113.04
% Change	-52.18%	-28.90%	-70.92%	-33.92%	-39.89%

Note. *As of January 20th, 2009. **As of January 20th, 2010.

Conclusion

Learning effective collaboration techniques prepares graduates for careers in industry. Part of the discipline of effective collaboration is to produce knowledge artifacts such as informational web pages, technical specifications and documents, and software code. We adopted an integrated wiki and SCM tool, Trac, to provide a resource to our capstone design students for managing their collaborative project data. Assessment of usage patterns from 2008 and 2009 indicates that project teams are making better use of the SCM tool, Subversion, to manage their document libraries. We are observing more usage of the Subversion tool to manage document creation and posting, and less usage of the more cumbersome and limited wiki-attachment approach.

Adapting the Subversion and Trac tools for use with the IPPD program has fulfilled immediate needs for remote management and sharing of team data. Future work will involve evaluation of more robust, possibly enterprise-quality collaboration and document management tools.

References

1. Stanfill, R.K., Wiens, G.J., Lear, W.E., Whitney, E D., "Institutionalized University and Industry Partnership in Multidisciplinary Design and Build: Product and Process Realization," Proceedings of the 2001 ASME International Mechanical Engineering Congress and Exposition, November 11-16, 2001, New York, NY, 11 pp. (CD-ROM, Book No. I00517).
2. Fitz-Coy, N., Mikolaitis, D.W., Stanfill, R.K., Vu-Quoc, L., "Maintaining Industry Partnerships in Integrated Product and Process Design Education," Proceedings of the American Society for Engineering Education 2002 Annual Conference & Exposition, Montreal, QC, June 16-19, 2002, 13 pp. (CD-ROM).
3. WikiMatrix. (January 21, 2010). WikiMatrix – Compare them all. In *WikiMatrix – Compare them all*. Retrieved January 21, 2010, from <http://www.wikimatrix.org/index.php>
4. MediaWiki.org. (October 8, 2009). Project:About - MediaWiki. In *MediaWiki*. Retrieved January 21, 2010, from <http://www.mediawiki.org/wiki/Project:About>
5. Edgewall Software. (December 21, 2009). The Trac Project. In *Edgewall Software*. Retrieved January 21, 2010, from <http://trac.edgewall.org/>.
6. TracHacks. (December 12, 2009). TracHacks – Plugins Macros etc. In *TracHacks – Plugins Macros etc*. Retrieved January 21, 2010, from <http://trac-hacks.org/>
7. Internet2. (n. d.). About Shibboleth. In *Shibboleth, A product of the Internet2 Middleware Initiative*. Retrieved January 21, 2010, from <http://shibboleth.internet2.edu/about.html>
8. Tigris.org. (n. d.). subversion: Subversion Features. In *subversion.tigris.org*. Retrieved January 21, 2010, from <http://subversion.tigris.org/features.html>