INMM 2007 Oral Presentation – Paper Number 291

Collaborative Software Tools for Safeguards Applications

Q. S. Bob TRUONG
Canadian Nuclear Safety Commission, Ottawa, Canada

N. HERBER
Eton Systems, Ottawa, Canada

Abstract

As part of its mandate, the Canadian Safeguards Support Program (CSSP) is responsible for providing assistance to the IAEA in knowledge sharing and knowledge transfer activities. To facilitate these activities, the CSSP has used various collaborative software tools. These tools are often referred to as “groupware” and most use some concept of a common workspace for the group members.

The CSSP has deployed several pilot and demonstration projects using these tools. These projects have been described in papers presented by the authors at previous INMM meetings. *Using Wikis as a Low-Cost Knowledge Sharing Tool* (presented in 2005) documented a very successful application of wiki technology to training development. Similar tools such as blogs and bulletin boards have also been discussed.

The successful deployment of some of these tools in pilot projects raises two questions:

(i) Do these tools have wider applicability in organizations such as the IAEA?

(ii) If a tool is to be deployed, should it be a commercial product or an open source product?

The answer to the first question is yes, these tools could be used to great advantage elsewhere within the Agency. The answer to the second question is not as clear, and in this paper, the authors explore the pros and cons of both commercial and open source tools.

The suggestions and conclusions in this paper will be of value to any organization considering deployment of collaborative software tools.

1 Introduction

The Canadian Safeguards Support Program (CSSP) provides support, documentation, and training assistance to the IAEA for Canadian manufactured safeguards equipment and related software tools. To facilitate these activities, the CSSP has used various collaborative software tools – the prime one being a wiki used for training material development. The wiki used for this purpose is “open source” software and the question arose: “What are the relative merits of open source versus commercial collaborative software?” This paper attempts to answer that question.

2 What is Collaborative Software?

At the most basic level, collaborative software provides a shared electronic workspace where co-workers or group members can develop, store, and share information and knowledge. Collaborative software is sometimes called a “workgroup support system” or “groupware” because it behaves like one piece of software that an entire group uses. Collaborative software lets users share files, notes, messages, etc. with others. It allows group members to comment on existing contributions, to create new content, and to edit existing content.

More traditional software that is used in a collaborative environment includes items such as shared calendars, email lists, network files stores, and web sites. However, this software, and other more traditional information sharing techniques, do not provide all of the advantages possible from a true collaborative workspace. The advantages and disadvantages of traditional sharing techniques are summarized in Table 1 below.

Table 1 - Traditional Sharing Methods

Method	Advantages	Disadvantages
Face-to-face meetings	- immediate feedback - information rich	- no history (unless a formal meeting with minutes) - no sharing possible after the fact
Phone meetings	- information rich - spontaneous	- no history - no sharing possible after the fact
Email	- universal access - “personal” history	- history local to each user - lots of “chaff” - little or no structure
Web sites (corporate intranets)	- central - universal access	- central control only - “broadcast” mentality - little user contribution
Network file stores	- central - comprehensive - long-term	- difficult to add contextual data - overwhelming volume

By examining Table 1, it is possible to infer many of the desirable characteristics of collaborative software (CS).

- CS should provide universal access with an easy to understand interface. Given the universal nature of the web, CS that uses a browser as a client is far preferred to CS with a proprietary client.
- CS should provide immediate feedback to contributors such that they can see their contributions without having to wait for a web master or system administrator to approve or post them.
- CS should allow easy user interaction. Users should not have to learn new workflows or complex command sequences to do things they already accomplish with tools such as word processors, email, and electronic calendars.
- CS should allow collaborative development of assets, which may be text documents, lists, plans, spreadsheets, and so on. Because the asset pool may be very large, some kind of search mechanism should be provided. In order to organize the assets, some kind of sub-workspaces should be provided.
- CS should provide long-term, central storage of all the group assets. This allows new group members to access the assets without being sent each one individually. It also allows for central control of access (based on security profiles) as well as secure storage and backup.
- CS should keep track of the revision history of the assets as well as a record of the discussion surrounding the asset development. In many instances, the analytical process, the discussion, and the revisions that occurs during asset development provides as much or more information as the final asset itself. History allows individuals to examine why certain decisions were made – not just the end result.

Collaborative software exists at several different scales. Knowledge Management is a set of practices used to create and share knowledge. Integrated Knowledge Management software systems attempt to provide all the “bells and whistles” needed to manage knowledge within an enterprise. One of the earliest such software packages is Lotus Notes. Integrated Knowledge Management software systems provide central storage for group assets with universal access and long-term storage. Such systems can be quite large and may require significant infrastructure to support them. They may require changes to traditional workflows to be used to best advantage. For many small organizations, the learning curve for users combined with the cost of the software and its infrastructure can be a significant impediment to its adoption. To put it simply, it costs too much, it takes too long, and it changes the organization.

One well-known groupware product is Groove. Initially developed by Groove Networks (a company founded by the originator of Lotus Notes), it was acquired by Microsoft for an undisclosed sum in 2005 and renamed “Microsoft Office Groove”. Groove differs from other groupware in several technical details – each user has their own local copy of the entire shared workspace and Groove uses proprietary peer-to-peer networking to synchronize the user copies.

Groove provides many features that can be delivered by Windows SharePoint Services, another Microsoft product intended to provide the infrastructure for developing enterprise-wide information portals. There are many different versions of SharePoint available, and each is preconfigured for a certain series of tasks. Some of the SharePoint versions can be very expensive to license, but WSS 3.0 (a limited functionality version of the larger packages) is available for free.

On a smaller scale, collaborative software such as blogs, message boards, or wikis can provide significant advantages to organizations without imposing a strain on either the budget or the delivery infrastructure. A thumbnail comparison of these three is provided in Table 2, below.

Table 2 - Comparison of blogs, message boards, and wikis

Feature	Blogs	Message Boards	Wikis
Description	Looks like a normal website or portal	The electronic equivalent of an office bulletin board	Looks like a normal website or portal
Edit or add content?	With a browser	With a browser	With a browser
Organization	- generally chronological - article based	- subdivided into forums (discussion areas) - each forum can contain many topics (also called threads) - each topic has replies	- individual web pages - groups of pages - hyperlinks between pages - plugins can be used to add blog and message board features
User interaction (depends on user group)	- add comments to existing articles - create new articles - edit existing articles	- create new forums - add new threads to a forum - reply to a thread	- create new pages - link to existing pages - edit existing pages
Attachments	- connect files to articles - place pictures in articles	- generally limited file upload capability - place pictures in replies	- connect files to pages - place pictures on pages - link pages to external files
Advantages	- central - universal access - excellent format for ongoing journals	- central - universal access - lots of user interaction - excellent for "conversations"	- central - universal access - lots of user interaction - history function tracks all changes and allows easy reversals
Disadvantages	- chronological organization can be limiting - generally a "broadcast" model with commenting allowed	- rambling discussions can make it very hard to discern consensus	- requires greater co-operation from users - can become unwieldy unless "subject owners" keep things tidy
Open source example	WordPress	SMF (Simple Machines Forum)	PmWiki (biggest wiki is probably Wikipedia)
Server resources required	- Apache-compatible web server - PHP scripting - MySQL-compatible backend database	- Apache-compatible web server - PHP scripting - MySQL-compatible backend database	- Apache-compatible web server - PHP scripting - flat-file storage

Blogs provide an excellent medium for recording journal-like observations and actions. Because blogs have a chronological organization, they are well suited to daily or weekly team newsletters. Blogs also provide a commenting capability so that team members can make observations about or add updates to the various postings.

Message boards are ideal for several streams of concurrent discussion. Message boards are organized by “threads” (topics) and are used by many organizations as a convenient way to manage help-desk queries.

Wikis are best suited for the collaborative development an intranet sites. Using just a browser, users can read pages, edit pages, create new pages, and leave commentary. A new user arriving at a page sees the current version – one that has been generated by the consensus of the contributors. They can also click on the “history” button to see the debate that went into the formation of the page.

Blogs, wikis, and message boards all use so-called “pull” technology – the user must visit the site to pull off new information. However, email notifications and RSS feeds (features available on all three) can convert them to “push” sites. For example, a user can request email notification when someone replies to their message-board posting or they can see all the new blog postings aggregated in their RSS reader.

3 Open Source Software

Wikipedia defines open source software as “computer software whose source code is available under a license (or arrangement such as the public domain) that permits users to use, change, and improve the software, and to redistribute it in modified or unmodified form. It is often developed in a public, collaborative manner.” For a long list of open source software, see http://en.wikipedia.org/wiki/List_of_open_source_software_packages.

Open source software is not necessarily free. An example of open source software is the Linux operating system. You can freely download the source or compiled versions. You can run it on any number of machines without paying any licensing fee. However, companies such as Red Hat use a subscription model to sell programming services, training, and support for their version of Linux.

By contrast, commercial or closed source software is sold under licensing conditions that generally forbid the end user from modifying the software for their own purposes other than through the manufacturer supplied configuration options or by purchasing additional plug-in modules that enable new functionality. Users are forbidden from redistributing the software to others – even within their own organization. In many cases, not only is the source code unavailable, but the data generated by the application is stored in a proprietary format, making it difficult to switch vendors.

Enterprises may be reluctant to use open source software for a variety of reasons. Although the low or non-existent cost may be attractive, commercial software has several apparent advantages:

- dealing with a commercial entity
- support
- training
- future roadmap

But in fact, these advantages are offered by many open source software packages. Linux has several commercial distributors (not just Red Hat). The open source blog software WordPress is available in either a free version or as a fully supported commercial product.

In addition to the support and training available from the commercial vendors of open source software, there are many consultants and developers available to provide training and support for these products. Certainly you need to choose a product that has a large enough user base to have attracted these individuals, but in many cases they can provide more responsive support than a larger vendor. Plus, since the source code is freely available, these consultants can modify the software to suit your needs – something that is unlikely to happen with commercial software unless your needs are shared by hundreds of other users.

Many open source software packages (such as the wiki PmWiki) have well documented future roadmaps, showing how they intend to develop and improve the software.

4 Security

All organizations have information that is private to various degrees, and for any collaborative software tool to be adopted, it must be able to secure the assets appropriately.

Regardless of the physical and technological mechanisms in place, security depends ultimately on the end users. Organizations need to establish protocols and procedures that help people to determine how to classify and store assets. For example, if a message board is being used for collaboration with individuals outside the organization, then the users must realize that all items appearing on the board will be public.

The next layer of security can be supplied by the collaborative software itself. It can restrict the access of certain users to certain areas and limit the functions available to them. For example, a small number of users may be designated as editors (able to change the content) while the remainder are designated as readers. Note that this restriction relies upon user authentication provided by a higher-level function.

Where the collaborative software is accessible from the internet at large, user authentication (verifying that a user is really who they claim to be) is generally is performed by the web server. Where the collaborative software is accessible only from within a firewalled intranet, user authentication can be performed by the network logon procedure.

5 Open Source Software Example

Since early 2004, the Canadian Safeguards Support Program (CSSP) and the IAEA have used an open source wiki (PmWiki) to share information during the development of training material for IAEA inspectors. The wiki has also been used for several other joint IAEA and CSSP projects. Support and hosting for the wiki is provided by Eton Systems. Since it was first installed, the software has been upgraded several times. These upgrades required no changes to the stored pages or files and the user interaction was unchanged.

Each project is assigned its own work area (known as a page group) with a home page and as many related pages as required for the project. Each of the authors who had access to the wiki has a personal profile page. Authors in Ottawa, Toronto, and Vienna are able to share files and change page content (including their profiles) as required. The pages have become mini project management centers with listings of milestone dates and progress reports.

Security for the content is provided by the web server – each author has a unique password. The wiki software detects the log on name of the author and automatically uses it to tag any changes the author makes. The wiki software also provides a "recent changes" summary that allows a quick review of items that have been updated.

Because a wiki is really a web site, it is possible to add links from many of the project pages to pre-existing content, such as file libraries. As of the writing of this paper, the wiki has over 20 page groups, some of which house multiple projects. Each user has a security profile that determines which of the page groups they can access.

The wiki software uses a plug-in model for adding extra features. On this wiki, modules for user authentication, email notification, user logging, and blog-style commenting have been installed. The wiki also uses a custom "skin" that governs the appearance of the entire website.

6 Commercial Software Example

Starting in November 2006, the CSSP and the IAEA examined several options for additional collaborative software applications. One of the most interesting items is a commercial package, Microsoft SharePoint Services. SharePoint provides an almost overwhelming number of infrastructure tools that can be used to build a complete collaborative software environment. Not only does SharePoint have blogs, wikis, and discussion boards, it also has files stores, picture libraries, lists, web sites, sub sites, RSS feeds, and integration with the Microsoft Office suite.

SharePoint offers one compelling advantages to the IAEA – the security model is integrated with the Windows security already deployed at the IAEA. This allows the IAEA to define user groups and permissions using tools that they have already deployed. Because SharePoint uses an SQL database to store all content, it is able to provide incredibly precise (or fine-grained) access control. For example, it would be possible to define the security level for each posting in a blog such that only those users with the correct credentials will see it when reading the blog.

Because this software is produced by Microsoft, they have designed it to interwork easily with the Office suite. In some respects, SharePoint looks like a networked version of the Office applications. It has a Project component that can be used for scheduling and planning. Many of the SharePoint components can directly import or export Office files such as Excel spreadsheets.

All of the interaction with SharePoint occurs within a browser (Microsoft Explorer 7 preferably). Users with administrative rights can create new web sites, blogs, and wikis with a few simple clicks. The SharePoint Designer package can be used to customize the look of pages, but most customization can be done using predefined "web parts".

The disadvantage to SharePoint is that it is a proprietary tool with a licensing model than can make it very expensive to deploy. SharePoint needs to run on top of the Windows 2003 server with Microsoft SQL Server 2005. None of these products are open-source.

However, the scalability of the product is impressive, and for an enterprise the size of the IAEA it presents an attractive platform to deliver collaborative software resources. It is definitely not just an "open the box and use it" solution, but it has enough scope to allow implementing systems that closely match the needs of the IAEA.

7 Concluding Remarks

Clearly, the benefits provided by collaborative software can help to improve the efficiency and effectiveness within small units and larger divisions of the IAEA. While the scope of the tasks faced by the Agency seem to point towards commercial software as the only choice, our experience has shown that even small implementations using open source software (such as the CSSP wiki) can provide usable results as soon as it is installed. Open source software is an excellent choice for small collaborative software pilot projects, but it can also be used to run very large and complex sites, such as Wikipedia.

Extending the benefits to other parts of the Agency in a consistent and well-managed manner requires something more than simply replicating the first success several dozen times. But the danger faced by the IAEA (or any large organization) is that the scope of an all-encompassing collaborative software system is so large, and the number of competing goals and priorities so numerous, that the implementation can get bogged down in "analysis paralysis".

One promising alternative for the Agency is SharePoint. Although it is commercial software with restrictive licensing, the licensing does not limit the applications developed, the web sites deployed, or the blogs created using SharePoint. SharePoint would allow the IAEA to bring collaborative sites on stream very quickly for ad-hoc use by small groups while simultaneously developing full collaborative applications to be used by a much larger audience. The large community of SharePoint developers and consultants can be drawn upon for help, training, and support. SharePoint's fine-grained security control would allow the Agency to organize and deploy their knowledge assets as required while keeping those assets secure.

We invite anyone interested in further information about these topics to visit our wiki at http://saturn.eton.ca/. It features live web links, contact information, and expanded discussion of this paper.

ACKNOWLEDGEMENTS

The authors wish to thank Ken Desson, Androcom Interactive Media, Ottawa, Canada, for his advice and assistance in preparing this paper. We are also grateful to the many individuals at the International Atomic Energy Agency, Vienna, Austria, who discussed these topics with us.

REFERENCES

1) Truong, Q.S. (Bob); Keeffe, R.; Desson, K.; *Integrating Learning & Doing: Single-Portal Access to Instruction, Decision-Making Tools and Information Sharing*, paper presented at the INMM 44th Annual Meeting, Phoenix, AZ, USA, July 27, 2003.
2) Truong, Q.S. (Bob); Herber, N.; Barroso, H.; Liguori, C.; *Using Wikis as a Low-Cost Knowledge Sharing Tool*, paper presented at the INMM 46th Annual Meeting, Phoenix, AZ, USA, July 10-14, 2005.
3) A WikiWikiWeb site with live web links, contact information, and expanded discussion of this paper: http://saturn.eton.ca/ .