Axon – An Adaptive Collaborative Web Portal

S. Berhe[1], S. Demurjian[1], H. Ren[1], M. Devineni[2], S. Vegad[2], K. Polineni[2]

[1] *University of Connecticut, USA*

[2] *Serebrum Cooperation, USA*

{steve, solomon.berhe}@engr.uconn.edu

haiying.ren@pw.utc.com

{mahitha, vsushil, krishna}@ serebrum.com

Abstract

Collaborative web portals allow groups to author, create, update, and share content via easy-to-use web-based interfaces (e.g., MediaWiki, Sharepoint, etc.) From an adaptability perspective, such portals are limited in their authentication, mobile device access, multi-channel document delivery, and version management. For example, in wikis, security is often at two extremes: anonymous users (no authorization and read-only browsing) vs. super users (unfettered access and update). As a result, these limitations have slowed acceptance of such portals in commercial and governmental settings for true collaboration, e.g., a patient/provider portal for health care has stringent security (HIPAA) requirements for access and sharing of personal health data. In this paper, we report on our research and development effort of a collaborative web portal that encompasses adaptability at: the application level, the document level (authoring and viewing), the security and version management level, and the look-and-feel of the portal itself.

1. Introduction

Over the past decade, web portals have emerged as a viable solution for collaboration [6, 17], ranging from information repositories (e.g., WebMD and Wikipedia [24]) to full-edged authoring/document collaboration. In this latter case, for registered users, these web portals provide a means to author, create, modify, and track documents of all types. A registered Wikipedia user can create new and modify existing content; this is a double-edged sword, since once given access, it is very difficult to control content, e.g., Wikipedia and other sites are rampant with factually incorrect content. In fact, freeware/open source products (e.g., Mediawiki [23]) or a commercial solution (e.g., Microsoft's Sharepoint [7]) allows any individual with sufficient expertise to generate their own web portal to meet special purposes and needs.

However, from an adaptability perspective, these products are not very granular in the level of customized security, navigation, access via mobile devices, advanced graphical user interface (GUI) features, dynamic user turnover, and multi-channel document delivery [8, 10, 13]. Consider health care, in which there are stringent HIPAA requirements [11, 22] on the security and exchange of data. In order to adapt a web portal or wiki to allow clinical researchers to collaborate to study diabetes in a patient population, these HIPAA requirements would dictate a higher level of security requirements than coarse-grained authorization and authentication (user names and passwords) typically offered by wikis/web portals. Moreover, current collaborative web portals are usually limited to web-based access. Continuing with the example, clinical researchers are dependent on study participants (patients) providing their glucose level on a regular basis. As not every patient is flexible in terms of accessing a desktop browser, additional access via mobile devices is highly desirable.

In this paper, we report on our research and development effort on applying adaptability to web portals for collaborative software requirements elicitation [19]. In this effort, the Axon wiki [25] offers document authoring, collaboration, publishing, versioning, and other capabilities, and has been prototyped with role-based access control (RBAC) [3, 18] at the application level, the document level (authoring and viewing), and the look-and-feel of the portal itself. These reasons were part of the main motivation to develop the comprehensive full-function

collaborative web portal in Axon, which significantly extends a wiki to adapt to the needs of enterprise users.

The work presented herein is similar to a growing body of research in adaptable web applications from varying perspectives. From a user-interface perspective, work in [5] presents an approach to separate the business related portion of a web application by using component-based XML documents to encapsulate layout and behavior on multiple abstraction levels to support flexible reuse. For example, user interfaces can be customized dynamically to meet various requirements without changing business rules. From a server perspective, the goal of work in [9] is to develop a J2EE-based Adaptive Server Framework to separate server, adaptation, and logic behavior. In this work, adaptation includes the consideration of network bandwidth into the page layout. As in our work, in [12], the problem of automatically organizing the structure and navigation of a web page based on the user, role, or group membership is addressed. Using text and usage mining, they propose a framework, its architecture and implementation to achieve self-adaptive web sites. Lastly, in [14] adaptability from an information-delivery perspective is considered by proposing a framework that adapts to the user's individual needs with respect to time, resource, and service of delivery. Overall, work on adaptation from different perspectives has influenced our work in developing the collaborative Axon web portal.

The remainder of this paper is organized into four sections. Section 2 provides brief background on Axon's capabilities and architecture. Section 3 details adaptability assumptions and concepts which are specific to Axon but can be generalized to collaborative portals. Section 4 introduces limitations of Axon and describes future features to address issues in a dynamically changing user, role, document, and contract environment. Section 5 concludes this paper.

2. Background on Axon's Concepts

Axon [25], a combined document collaboration, content management and wiki, is a Java Enterprise Edition platform 1.4, AJAX-based [16, 21] web portal targeting enterprise adoption by offering a wide range of document authoring, publishing, versioning [13, 17], and other capabilities. Axon's advanced workflow allows enterprises to set up sophisticated business processes for content creation, editing, review, publishing and archival. The main intent is to provide a full-capability adaptable wiki with fine-grained RBAC that meets critical requirements in terms of security, flexibility, document delivery, and administration. This allows Axon to meet customized requirements of various application domains which are moving towards collaborative web portals. In Figure 1, the Axon architecture is illustrated. Axon uses a Model View Controller design that divides application logic into manageable units and promotes scalability.

The design of Axon is also standards-based: JCR compliant Document Management System (DMS); Hibernate to interface with any relational database platform; Lucene API for search; Directory Server for user information; Blowfish for data security; SSL for secure communication with the server; and, 'Echo' for the AJAX User Interface. The clients can either be connected via workstations, laptops, or mobile devices. The Presentation Layer provides the typical means to access the underlying application. BrainStorm, which is the name of our software requirements elicitation effort [15], contains two grayed boxes (application specific, changeable) and two non-grayed boxes (representing core Axon functionality). The Application Layer embodies many of the various underlying technologies to support Axon's core functionality. Lastly, the Data Layer allows Axon to be configured with any relational database as a backend.

Axon is a full-function wiki for content creation (WYSIWYG), document publishing (e.g., Web, PDF, RTF, etc.), document distribution (e.g., Email, Print, Fax, etc.), mobile access via a BlackBerry [4], and RBAC to allow collaboration among users that are sharing a particular task. As shown in the top portion of Figure 1, the Axon Wiki is loaded from a web server into a multi-framed structure that includes a number of features. First, there is a top bar of functions (box A with Hide, History, Import, Export, Email, and Print); tabs for Spaces and Index (box B) where the Spaces tab is organized as accordions (e.g., Sales and Marketing, Finance, etc.) with parent topics (Status Reports), child topics (e.g., 2008 and 2007) and grandchild topics (e.g., Driver testing, Driver review, etc. – in box C with icons to create, edit, etc. topics – in box D). The spaces, accordions and their topic trees are customizable based on domain. Second, there is a main window with TOPIC and DOCS (box E) tabs: the TOPIC tab shown selected in the top portion of Figure 1 is editable (XHTML) for the selected topic (i.e., AMSS Project Plan) with Edit, History, Intralink, etc. – box F; and, the DOCS tab shown selected in the bottom portion of Figure 1 tracks the attached documents (e.g., PDF, Word, MPEG, etc.) for the topic, with the ability to Attach, Copy, Paste, Cut, Check-Out, Check-In, Replace, Delete, History, and Email – box F.

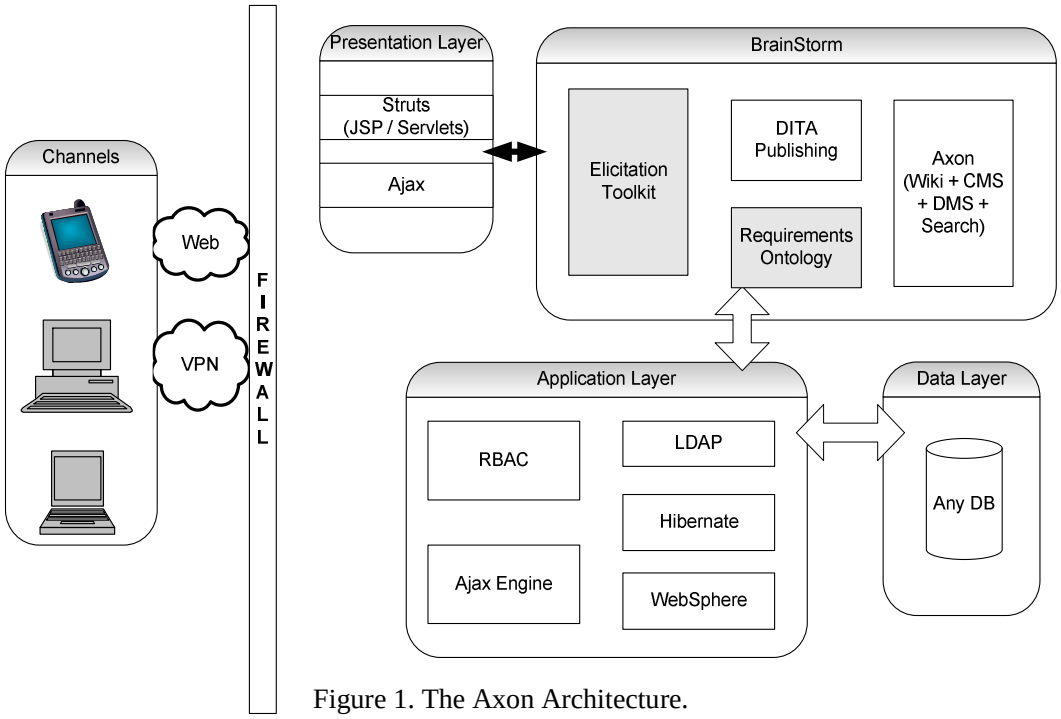

Figure 1. The Axon Architecture.

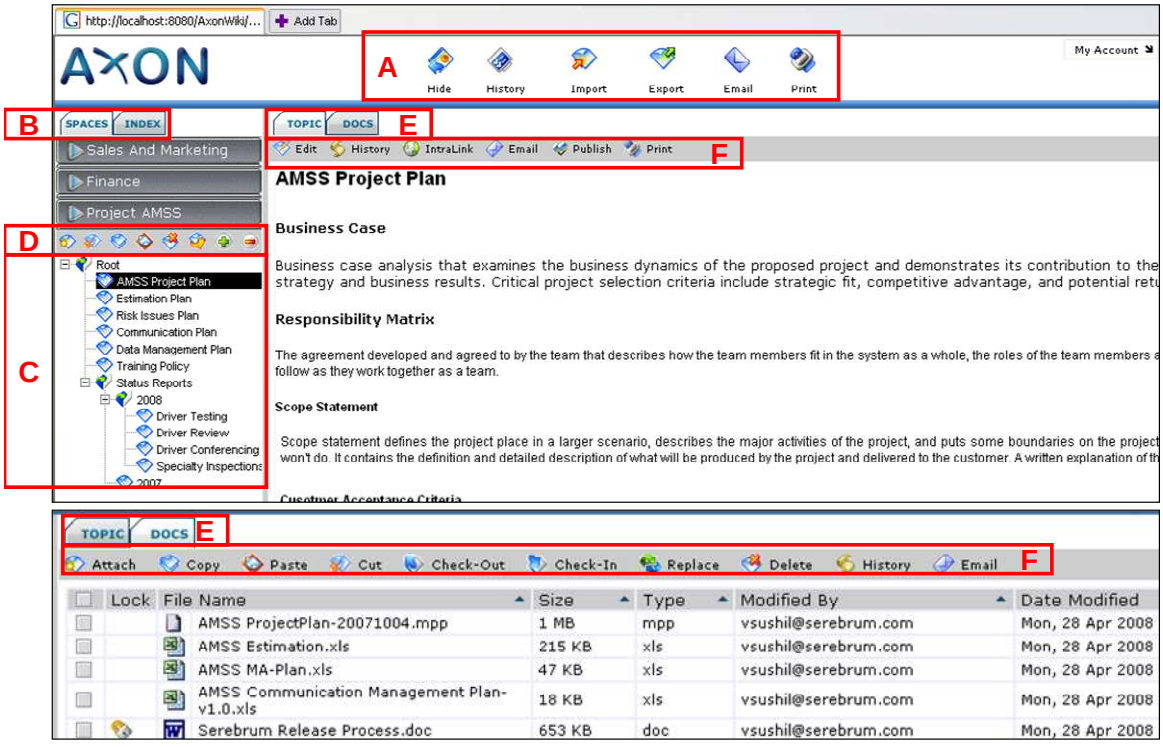

Figure 2. The Axon Wiki.

In Figure 3, additional capabilities of Axon are shown. At the top of Figure 3, the Ajax-based WYSISYG editor for creating and modifying XHTML document content is shown. At the bottom left of Figure 3, the document assembly feature is illustrated, which allows users to select documents of different types (e.g., Word, RTF, PPT, html, etc.) which are then combined to form a resultant document in different formats (e.g., PDF, RTF, HTML, etc.). At the bottom right of Figure 3, Axon for a BlackBerry [4] is shown, which allows limited viewing/editing of content.

3. Axon's Various Adaptation Approaches

In this section, we explore three adaptation capabilities of Axon: two functional qualities and one quality related to customization. . The first functional quality, in Section 3.1, is related to two of Axon's Wiki security capabilities: role-based access control (RBAC) [3, 18] and document change tracking capabilities. In terms of RBAC, Axon transcends the typical on-off capabilities of most portals with fine grained, changeable privileges. In a health care setting, access to patient data is very dynamic, when one considers physicians covering for one another (nights, weekends) and emergent situations; role replacement and privilege modification are both needed in this setting. In terms of document tracking, a mechanism that tracks all past versions will be critical in a health care setting, to allow historical patient data to be maintained. For the second functional quality, in Section 3.2, we illustrate the leveraging of Web 2.0 technologies, namely, AJAX, to improve usability. Typically, registered users are able to create and edit content using simple plain text, html, or wiki markup editors; as a result, users access and modify information via a desktop web browser.

However, as more dynamic and data sensitive domains such as health care target collaborative web portals, content accessible via a browser in an html format is limiting. Emerging Web 2.0 technologies can play a pivotal role in dealing with this limitation. Consequently, in Section 3.3, we explore the quality related to customization by detailing Axon's approach to project management and discussing various document deliveries strategies. Current open source or commercial collaborative web portals provide the storage of a wide variety of document content. A logical next step is to allow the adaptation of this content, allowing the content (from different sources) to be combined in different ways, providing a high degree of versatility in content management.

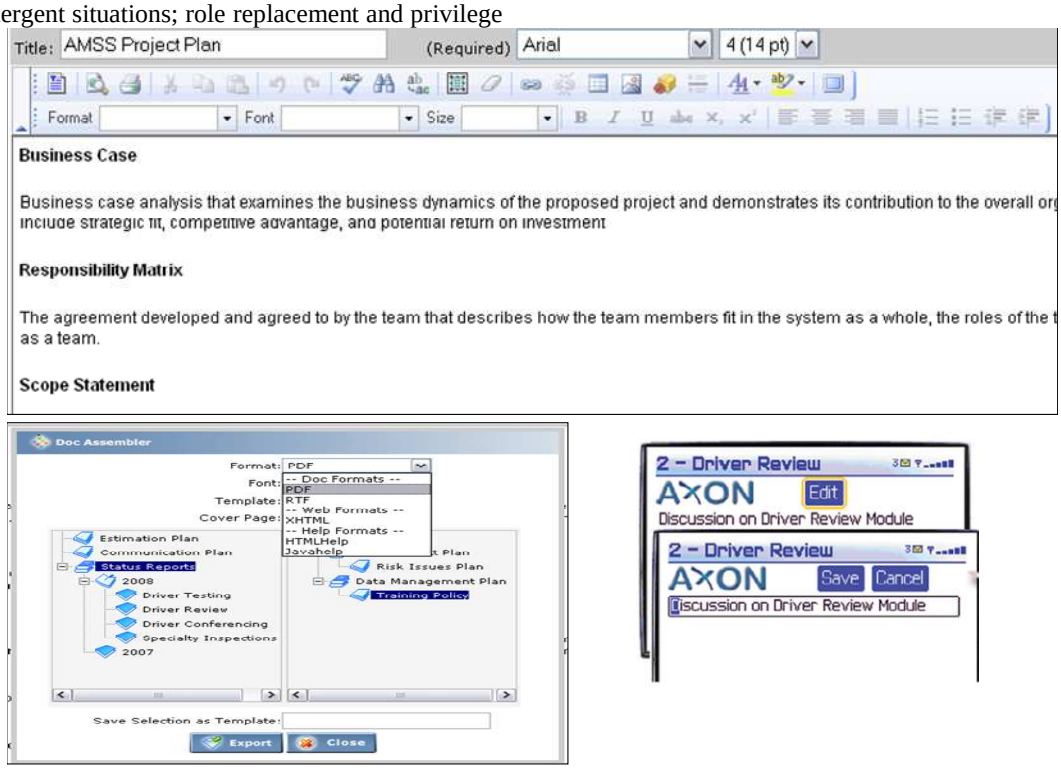

Figure 3. Additional Axon Capabilities.

3.1 Customized Security/Version Management

Most wikis protect content via authorization and authentication. For example, in Wikipedia, you must be a registered user in order to create or edit content. But, in many cases, the access is all or nothing; fine grained control of content is not supported. To explore this issue, in Figure 4, the security capabilities of Axon are summarized with typical roles illustrated (e.g., Guest, Author, etc.). Axon's security implementation uses Sun's Java System Access Manager which provides scalability, single-sign on, centralized application administration, and other important features. In a Accordion (e.g., Sales and Marketing, Finance, etc., in Figure 2), the content control is achieved for parent, child, and grandchild topics; in Figure 4, the user by role can be given privileges to View and/or Edit each of these topics, and for the TOPIC and DOCS tab, be given specific access to individual buttons (box F in Figure 2; for TOPIC – Edit, History, IntraLink, etc., and for DOCS – Attach, Copy, Paste, etc.). Users can also be authorized to multiple roles, but are limited to playing a single role in a session. The permissions, as shown in Figure 4, define whether a user (by role) can have access to the buttons of the Global (box A in Figure 2) and the Tree (box D in Figure 2) Menus. All of the permissions that control access to the look-and-feel of Axon are represented in Figure 4 with the Yes/No access to the Global and Tree Menus, and TOPIC and DOCS Buttons. As a result, when the user logs on to Axon with a role, the entire look-and-feel and the content is customized.

This fine-grained and adaptable security capability positions Axon as a viable alternative for sensitive domains such as the health care. Unlike open-information sharing portals (Wikipedia), a health care portal would not provide free-and-unfettered access to all information; patient education materials would be available, but if the portal offers personal health record capabilities, the associated data would need to be strictly protected (Sarbanes-Oxley/HIPAA conforming [11]). The advantage of using Axon for this application would be to facilitate the exchange of information among patients and providers, given that these web portals are accessible from anywhere at any time. Unfortunately, current implementations of wikis are not designed with a fine-grained access control mechanism. Axon addresses this issue with the help of its adaptable RBAC matrix (Figure 4) that allows the administrator define which actions may be performed by which user (by role) during which period on which project and document. Thus, Axon could be used by clinical researchers in the health care domain to exchange, publish or collaborate on a topic, which is only intended to be read by a limited audience. In Axon, default roles are Administrator, Manager, Author and Guest; these can be customized according to the domain in which they will be used, e.g., Provider, Researcher and Patient. On the application level, permissions are enforced by displaying only buttons and projects, for which a role is authenticated.

Finally, to protect the system against authenticated users with malicious intention, Axon and other collaborative web portals provide document history [17] to track all document changes (time, user, document, action). In addition, all folders and documents have the ability to be configured for email notifications, so that users can be alerted when any document is downloaded, uploaded, versioned, edited in any way etc. Especially, in critical domains, this must be provided to satisfy HIPAA regulations [11]. Thus, any malicious action can be repaired with a rollback function that restores the document to the last working state. Or, alternatively, all changes to patient data can be carefully tracked with versions to create a history of patient care.

3.2 Adaptation via Web 2.0 Technology

Modern collaborative web portals must not only concentrate adaptability on various ways of accessing them or on the security policies required by a domain, but also adapt to users with limited experience (e.g., older patient participating in the diabetes studies, computer-phobic physicians, etc.). Therefore, users should be able to utilize the full functionality as intuitively as possible. As many users are familiar with desktop applications such as Microsoft's Word, Axon provides true WYSIWYG editing and enhanced navigation features such as cut & paste and drag & drop (Figure 3). In addition, full search of document content is provided. Search results can be sorted, and the high granular access control ensures that users can only search content within the Accordions to which they have been authorized. Users have Simple' search option and an 'Advanced' search option. The Advanced search option provides different search parameters (e.g., topic, author, file type, date, etc.). To improve performance, each topic at every level has a unique index/location and is sorted by frequency interval. All of these features improve the user's experiences, and are implemented using Web 2.0 technology AJAX [16]. Axon's editor and application interface design is based 100% on AJAX, fully leveraging features such as dynamic database-driven tree menus, and switching from view to edit modes.

Feature	Sample User Types --> Guest Permissions --> Read	Author Read, Write, Add, Create Topics, Manage Users for Created Topics	Manager Read, Write, Add, Archive, Create Topics, Manage Users for All Users in Accordion	Admin All of Manager + Create Accordions Global User Management
Global Menu (Box A)	Yes/No	Yes/No	Yes/No	Yes/No
Hide	Yes	Yes	Yes	Yes
History	Yes	Yes	Yes	Yes
Import	No	Yes	Yes	Yes
Export	Yes	Yes	Yes	Yes
Email	Yes	Yes	Yes	Yes
Fax	Only if assigned	Only if assigned	Can assign to accordion users	Can assign to all users
Print	Yes	Yes	Yes	Yes
Tree Menu (Box D)	Yes/No	Yes/No	Yes/No	Yes/No
New Topic	No	Yes	Yes	Yes
Copy	No	Yes	Yes	Yes
Paste	No	Yes	Yes	Yes
Rename	No	Yes	Yes	Yes
Archive	No	No	Yes	Yes
Accordion Topics (Box C)	View /Edit	View /Edit	View /Edit	View /Edit
Parent Topic	View	View /Edit	View /Edit	View /Edit
Child Topic	View	View /Edit	View /Edit	View /Edit
Grandchild Topic	View	View /Edit	View /Edit	View /Edit
Topic Buttons (Box F)	Yes/No	Yes/No	Yes/No	Yes/No
Edit	No	Yes	Yes	Yes
History	Yes	Yes	Yes	Yes
IntraLink	No	Yes	Yes	Yes
Email	No	Yes	Yes	Yes
Publish	No	Yes	Yes	Yes
Print	Yes	Yes	Yes	Yes
DOCS Buttons (Box F)	Yes/No	Yes/No	Yes/No	Yes/No
Attach	No	Yes	Yes	Yes
Copy	Yes	Yes	Yes	Yes
Paste	Yes	Yes	Yes	Yes
Cut	No	Yes	Yes	Yes
Check-In	No	Yes	Yes	Yes
Check-Out	Yes	Yes	Yes	Yes
Replace	No	Yes	Yes	Yes
Delete	No	No	No	Yes
History	Yes	Yes	Yes	Yes
Email	No	Yes	Yes	Yes

Figure 4. Adaptable Role-Based Access Control Security.

3.3 Adaptation in Topic Management/Delivery

Axon organizes content and topics as reusable pieces, namely "infolets", which can be assembled into one or more documents and published in any format (e.g., Web, PDF, RTF, etc.) as given in Figure 3. Any changes in an "infolet" are propagated in real time across all publishing channels [10]. Topics (Figure 2) are organized as accordions and their Topic trees are adaptable based on domain. Each topic (parent, child, or grandchild) can have a single XHTML document (in the main window) along with one or more other documents (e.g., PDF, Word, MPEG, etc.) that are attachable and accessible via the DOCS tab (see the bottom portion of Figure 2). The topic tree in conjunction with the TOPIC and DOCS tabs provides a high degree of flexibility when establishing permissions (what a user can and cannot do), allowing the tool to easily adapt to different user needs and deliver exactly enough content (and no more) to each user by the permissions assigned to his/her role.

Axon utilizes a tree structure to organize the folders in the 'spaces' on the left. Each folder can have editable text content (the top portion of Figure 2 – the TOPIC tab) and documents attached to it (the bottom portion of Figure 2 – the DOCS tab). When a folder is selected in the tree, the documents for that topic are displayed under the 'DOCS' tab on the right, as shown in the bottom portion of Figure 2. Documents of any type and size may be uploaded. In our example, a topic could be "The Study of Diabetes in a Patient Population" with the child topics "Experiment Specification", "Drug Discussion", "Treatment" and "General Discussion". Information updates can be converted to various formats the patients and researcher desire and delivered via Email, Print or Fax. Thus, Axon's topic-based approach erases the limitations of the document-centric approach to model information flow by separating the content creation and

publishing activities into a flexible framework so that information is updated in real-time across multiple channels on demand.

4. Limitations and Ongoing Work

While Axon [25] is a commercially product, there are limitations and enhancements that are still under investigation. First, in the current release, scalability has an impact on a number of features: system performance, and user, accordion, and topic document administration and their management. In its usage to-date, Axon's document and project management system has not as yet been stressed with a high volume of users (and associated documents), which may impact performance. Therefore, ongoing work is focusing on improving critical system design features. This includes the way that domains could be used, the way that folders must be organized and the structure of the entire system for locating topics and their associated documents. This may have a considerably impact on the response time, directly affecting actions such as when a user navigates between accordions and their topics (expanding the tree – see A in Figure 2).

Scalability can further impact system administration; realistic enterprise applications with thousands of accounts need to be precisely managed by a security officer or other administrative personnel. To address this issue, we are carefully re-designing the user administration design and architecture, so that system wide functions, such as setting up new users, resetting passwords, disabling accounts, delete topics and/or documents loaded by mistake, can be easily performed. Scalability at the user level must also be addressed. For example, a user may have tens of accordions and each of these accordions could have dozens of parent, child, and grandchild topics. When the user is interested in making this information available to other users by role, privilege definition and maintenance become monumental tasks, since permissions are at the topic level. Therefore, we are currently working on a mechanism that seamlessly facilitates project definition and the intricate privilege assignment to individual roles. For example, one approach under consideration is having permission-predefined topic trees that can be included in an accordion and then customized by name, privilege, etc. Finally, in the case where Axon is being utilized to manage a specific repository of shared documents for a defined timeframe, there must be a mechanism to archive and/or delete when a time limit is reached or an employee leaves. For example, at the end of a contract term, all of the documents and information stored on the system must be transferred or deleted automatically.

In terms of ongoing work, we note that Axon is currently being used by a diverse stakeholder team of faculty researchers, computer scientists (without health care background), and health-care providers to organize documents and collaborate in order to assist in the processing of making informed health-information technology (HIT) decisions. Specifically, the Ethel Dongahue TRIPP Center at the University of Connecticut Medical School [26] has an ongoing two year project to integrate HIT into seven community health centers. A vital aspect of this work is the evaluation and usage of a collaborative web portal (Axon) to facilitate this work by a user community (health care professionals) that is often averse to new computing technology and slow in its adoption. We expect to report on our experiences with Axon in a security-constrained setting (HIPAA) in the near future.

5. Conclusion

In this paper, we have presented the collaborative web portal Axon in Section 2, exploring its architecture (Figure 1), core functionality (Figure 2), and additional capabilities for content creation/management (Figure 3). Using this as a basis, in Section 3, we explored Axon's adaptability from three perspectives: the security and version management level, and the look-and-feel of the portal (Section 3.1); the content creation level via Web 2.0 technologies (Section 3.2); and, the document level with respect to authoring and viewing (see Section 3.3). We demonstrated all of these capabilities through the consideration of a health care domain, where secure and flexible access is a strong requirement. The end result is an adaptable framework, where an application domain is able to utilize Axon by customizing the portal to satisfy any unique application requirements. Since the work on Axon is continuing, in Section 4, we detailed limitations and potential solutions; we also briefly discussed an ongoing actual usage of Axon in a health care setting. We believe that the end result is a truly adaptable and customizable web portal that transcends available systems.

References

[1] M. Barla, et al., "Adaptive Portal Framework for Semantic Web Applications", *Proc. of ICWE 2007 Workshops. 7th Intl. Conf. on Web Engineering*, 2007.

[2] Y. Demchenko, et al., "Security Architecture for Open Collaborative Environment", in *Advances in Grid*

Computing - EGC2005, P. Sloot, et al. (eds.), Springer LNCS, Vol. 3470, 2005.

[3] S. Demurjian, et al., "User Role-Based Security Model for a Distributed Environment", in *Data and Applications Security: Developments and Directions*, B. Thuraisingham, R. van de Riet, K. Dittrich and Z. Tari (eds.), Vol. IFIP 73, Springer, 2001.

[4] S. T. Dunnavant, "Blackberries in Support of Technology", *Proc. of the 29th Annual ACM SIGUCCS Conf. on User Services*, 2001.

[5] Z. Fiala, et al., "Design and Implementation of Component-Based Adaptive Web Presentations", *Proc. of the 2004 ACM Symp. on Applied Computing*, 2004.

[6] A. Forte and A. Bruckman, "Constructing Text: Wiki as a Toolkit for Collaborative Learning", *Proc. of the Intl. Symp. on Wikis*, 2007.

[7] M. Gerow, *Creating Client Extranets with Sharepoint 2003*, Apress, 2006.

[8] M. Jones, et al., "Wikifying your Interface: Facilitating Community-Based Interface Translation", *Proc. of the 6th Conf. on Designing Interactive Systems*, 2006.

[9] Y. Liu, 2006. "Enabling Adaptation of J2EE Applications using Components, Web Services and Aspects", *Proc. of the 5th Wksp. on Adaptive and Reflective Middleware*, 2006.

[10] J. Maxwell. "Using Wiki as a Multi-Mode Publishing Platform", *Proc. of the 25th Annual ACM Intl. Conf. on Design of Communication*, 2007.

[11] R. Mercur. "The HIPAA-Potamus in Health Care Data Security", *Commun. of the ACM*, 47(7), 2004.

[12] A. Mikroyannidis and B. Theodoulidis, "A Theoretical Framework and an Implementation Architecture for Self Adaptive Web Sites", *Proc. of the 2004 IEEE/WIC/ACM Intl. Conf. on Web Intelligence*, 2004.

[13] A. Mørch,. "Three Levels of End-User Tailoring: Customization, Integration and Extension", in *Computer and Design in Context*, M. Kyng and L. Mathiassen (eds.), MIT Press, 1997.

[14] L. Nevile, "Adaptability and Accessibility: A New Framework", *Proc. of the 19th Conf. of the Computer-Human Interaction Special Interest Group of Australia on Computer-Human Interaction: Citizens Online: Considerations for Today and the Future*, 2005.

[15] P. Pia, et al., "BrainStorm: Collaborative Customer Requirements Elicitation for Distributed Software Development", *Proc. of 31st Annual Software Engineering Workshop*, 2007.

[16] A. Puder, "A Cross-Language Framework for Developing AJAX Applications", *Proc. of 5th Intl. Symp. on Principles and Practice of Programming in Java*, 2007.

[17] M. Sabel, "Structuring Wiki Revision History", *Proc. of the Intl. Symp. on Wikis*, 2007.

[18] R. S. Sandhu, et al., "Role-Based Access Control Models", *IEEE Computer*, 29(2), 1996.

[19] NSF Award #0611053 to Serebrum, Inc., SBIR Phase I: BrainStorm - Collaborative Customer Requirements Elicitation for Distributed Software Development.

[20] W. Xiao, et al., "Online Collaborative Software Development via Wiki", *Proc. of the Intl. Symp. on Wikis*, 2007.

[21] http://developers.sun.com/ajax

[22] http://www.hhs.gov/ocr/hipaa/

[23] http://www.mediawiki.org/wiki/MediaWiki

[24] http://www.wikipedia.org

[25] http://www.serebrum.com/axon/index.html

[26] http://trippcenter.uchc.edu/