Evolution in Clinical Knowledge Management Strategy at Intermountain Healthcare

Nathan C. Hulse, PhD[1,2], Joel Galland, MS[1], Emerson P. Borsato, PhD[1],
[1]Intermountain Healthcare, Murray, UT; [2]Department of Biomedical Informatics, University of Utah, Salt Lake City, UT;

Abstract

In this manuscript, we present an overview of the clinical knowledge management strategy at Intermountain Healthcare in support of our electronic medical record systems. Intermountain first initiated efforts in developing a centralized enterprise knowledge repository in 2001. Applications developed, areas of emphasis served, and key areas of focus are presented. We also detail historical and current areas of emphasis, in response to business needs.

Introduction

The amount of information and knowledge processing required in modern medicine is staggering. As the pace of medical discovery accelerates, systems that facilitate the capture and distribution of clinical knowledge will only become more important for organizations who champion the distribution of clinical best practice in caring for their patients. The key goal of any clinical knowledge management system should focus on the conversion of 'tacit knowledge' (subjective, cognitive knowledge often stored in the minds of clinical caregivers) to 'explicit knowledge' (knowledge which has been codified, verbalized, or articulated in such a way that facilitates sharing) [1]. From a purely organizational perspective, efforts in knowledge management are focused toward retaining one of the most valuable resources available to it: the collective knowledge of its workers [2].

Knowledge management (KM) systems have become a key component in the architecture of modern electronic medical records. They play an important role in that they control the authoring, review, publication, delivery, and versioning processes that surround clinical knowledge assets for consumption within these frameworks. Examples of clinical knowledge can be highly diverse, including content such as order sets, decision support rules, nursing protocols, care guidelines, and education resources. The intended targets for these clinical knowledge assets can be human-focused (as in the case of narrative clinical care guidelines), computer-focused (rule bases for consumption by inference engines) or aligned for consumption on both ends of the spectrum. For consistency, when we refer to the phrase "clinical knowledge management", we mean the "entire process by which clinical knowledge is created, made available, and maintained within an EHR system" as defined in a recent study by Sittig et al [3].

Although informatics tools and excellence in medical informatics at Intermountain Healthcare date back to the work of Dr. Homer Warner in the 1960's, our efforts in the area of clinical knowledge management have been much more recent [4-5]. To better address the need for clinical knowledge transfer throughout the company, Intermountain has adopted a strategy in the past 11 years that focuses on creating infrastructure, tools, content creation teams and organizational support for the development of a clinical knowledge repository. In this paper, we will explore both the historical direction of the group, an overview of progress made to date, and current strategic and functional objectives.

Background/Case Study

Intermountain Healthcare is a not-for-profit integrated delivery network consisting of hospitals, health plans, and affiliated physicians that serves the Intermountain West [6]. It employs more than 32,000 individuals (including over 700 clinicians), and provides a full range of medical services to communities throughout Utah and southeastern Idaho. It operates 23 hospitals and more than 165 clinics throughout the region, and either owns or supports 19 community clinics serving uninsured and low-income patients[10].

Early stages of KM at Intermountain

The KM development process started in earnest at Intermountain in 2001. At that time the efforts were driven to build an environment capable of store, retrieve, compose, and review structured knowledge assets. The system structure designed to accommodate the KM is composed of four primary components:

(i) **Database and Backend Services:** At the heart of Intermountain's Knowledge Repository (KR) is an Oracle database with a series of supporting services designed to store and retrieve various types of knowledge artifacts, (there are many types supported, but the primary types include Binary Long Objects (BLOB), Character Long Objects (CLOB) and XML documents.

(ii) **Publication and content retrieval services:** A service layer called ROOK (Ready Only Online KR) which is a service designed written in Java that reads/writes the content from the KR database and delivers it, typically through an HTTP URL request.

(iii) **Structured Content Authoring Tool:** Intermountain has developed and maintained a browser-based authoring tool known as the Knowledge Authoring Tool (KAT) that enable end users, generally healthcare professionals, to compose structured knowledge assets (based on XML technology) without the need for any technical understanding of data models, schema validation or structured syntax [7].

(iv) **Knowledge Repository Browser:** This tool was the first user-facing tool developed by the KR team that allows users to interact with content in the repository, based off the users' role. The Knowledge Repository Online (KRO) tool enables users to search, browse and submit reviews to its authors [8].

This is the basic structure that was architected and placed in production between 2002 and 2004 and it provides knowledge resources to the end users as well to other systems and modules. In many cases, end users are creating, updating and consuming structured knowledge content from the KR through other internal client systems without necessarily direct interaction with KR front-end applications. The KR codebase includes an API that facilitates these routines, including publication, versioning, content model validation, and distribution.

Knowledge Asset Structure

Intermountain's KR database was designed to store knowledge content and keep multiple versions of it. Content is never deleted from the repository, making the 'lifecycle' of content development and evolution fully traceable over time. Each knowledge asset is composed of two parts, stored as related documents. Our high-level architecture follows the clinical document architecture (CDA) paradigm in that each asset has both a document header and a document body. The document header is standardized across the entire repository, regardless of content type or category. Within it, a document's metadata is stored, containing key pieces of information, including the document title, the document identifier, key words, change history, custodians, publisher, etc. The document's header metadata is structured in a XML format and all documents in the repository respect the same metadata information model. The document body represents the document itself and it can be of nearly any format like images, Microsoft Word documents, spreadsheets, Java rulebases, XML documents, etc.

KR documents are divided in document categories. A category defines the purpose of document and its type of asset. A document belongs to only one single category (e.g. order set, discharge instruction, image, clinical procedure, etc.).

Given the various usage scenarios supported by the KR, it is valid to say that KR hosts both structured and non-structured documents. Structured content is primarily stored as XML documents that have a XML schema definition that details the information model associated with that category. A structured XML category typically also has the following XML supporting documents:

- **XML Schema (XSD file):** The schema is used to validate the knowledge asset in order to ensure that all the minimum requirements in terms of structure, datatypes, cardinality, and content organization for the document are met.
- **XML Transformation (XSLT):** This is used to transform the XML document into more consumable formats. In the case of human-focused targets, the output is generally HTML, but it can also be PDF or spreadsheets. In many cases the XSLT transformation also injects some JavaScript codes that can turn presented documents into smart forms and even small applications [9-10].
- **XML Template:** This type of asset defines a 'start point' from where new documents are created; it generally defines the main items required for a document in order to assist the user in the document creation process.

- **Electronic authoring form:** This asset is consumed by KAT and represents a user-friendly form capable of creating an XML document instance based in a XML schema through a web application. This feature enables end users to create XML structured knowledge assets without requiring any technical knowledge of XML sytnax.
- **Validation Rules:** This is an optional external element (also defined in XML) that offers a cross-attribute and cross-element validation scheme that goes transcends the basic XML schema validation rules by mapping specific schema error codes to user-friendly phrases that assist users in identifying and correcting document issues related to schema validation [11].
- **Pre-processing document code:** This is a java compiled codebase that can be associated to a document category to pre-process documents at runtime. This Java code implements an interface and it is loaded on the fly. One of the types of usages of this file is that it can automatically detects the current document configuration and use it to automatically populate other parts of the document. This solution was designed augment the form-based content creation environment with programmable components.

Knowledge Content Lifecycle

The typical knowledge lifecycle at Intermountain can be summarized in a few key steps. These include knowledge asset design, authorship, review, publication, and updates. Early phases of content development at Intermountain have focused toward supporting existing content creation teams including specialty specific Clinical Programs (best practice care standardization teams), nursing education teams, and 'grassroots' content efforts. Several clinical teams and units worked closely with KR personnel to develop tools that were governed largely by processes and review cycles internal to the group.

In more recent years, several factors have converged, resulting in the need for a new set of workflow-driven review processes that support the knowledge content lifecycle. Local 'grassroots-based' content efforts have thrived largely on 'ad hoc' based content cycles. However, as it pertains to enterprise-level content creation and delivery, a more sophisticated set of systematized reviews has proved importance for compliance and regulatory reasons.

A key tool, which is currently under development to support this need, is the document review/approval tool. This has proved particularly important in the case of enterprise-oriented order sets that are being developed in lockstep with an internally developed computerized provider order entry platform. Given the enterprise wide focus of this content, we have developed a workflow driven tool that orchestrates several of these processes; automating required review and feedback from appropriate stakeholders before specific processes (activation and publication) can occur. In this manner, appropriate signoff from groups like compliance, nursing, pharmacy, and other divisions is required in to fully refine content prior to its use across the enterprise. All feedback gathered throughout these cycles is traceable and stored along with the versioned content in the knowledge repository.

Supporting Roles and Personnel

There are a number of important personnel roles that play a part in our knowledge asset development processes. These are intended to support accuracy and re-use throughout the effort. Locally, these types of staff include:

(i) *Knowledge Manager*: Primarily responsible to evaluate what are the available resources to design the structure of the knowledge, assign tasks and supervise the creation of the knowledge asset.

(ii) *Knowledge Engineer* (KE): Responsible for the design of knowledge asset information models, including all supporting documents. These include:

   a. *Asset Structure:* Typically an XML schema that represents the information model of the knowledge

   b. *Authoring Form:* User-friendly form for creation and maintenance of the knowledge, which will be saved in a XML format matching the schema definitions for that asset category.

   c. *Presentation:* Many of the XML-based knowledge assets are rendered as HTML at runtime for human readable consumption. To accomplish this, the KE is charged with composing XML transformation files (XSLTs) that transforms the XML content on the fly. This transformation file can be updated at any time to match new definitions or requirements.

(iii) *Clinical Experts* (CE): Typically healthcare professionals recognized as experts in a particular domain or discipline. They provide the KE the knowledge asset requirements.

(iv) *Content Writers*: Typically healthcare professionals that create content using the authoring tool and the forms designed the KE with the clinical experts. They are in charge of creating and updating content and publishing it in the KR.

(v) *Content Reviewers*: Users charged with the review of content written by the content experts in order to ensure its integrity and accuracy. They can interact with the content experts by submitting their review using structured review forms (which are often available via the process-driven workflow review tool).

(vi) *Content Approvers*: are people in charge of the final approval on the content written by the Content Experts).

(vii) *Content Consumers*: End-users that access the knowledge content through its service layer. They are healthcare professionals in general and/or patients

Infobutton Manager

The Infobutton Manager is another key component of Intermountain's KM infrastructure that was originally implemented by Del Fiol et al [12] based off reference implementations at Columbia University [13]. It is widely used for quick, point-of-case access to electronic reference information from within Intermountain's electronic medical record.. This project utilizes relevant contextual pointers that infer information needs as users interact with the system, and passes these parameters intelligently on to resource providers so as to provide relevant, contextual information with minimal user interaction. Intermountain distributes several types of knowledge content via this platform, including content that is created and managed in-house, as well as others that come from external resources via third-party providers.

The Infobutton Manager interprets the user's request for information and, based on certain parameterized context (e.g. patient age, provider role, pregnancy status, source concept, etc.), it selects the best source (eResource) of information to respond to the request. Then it loads the eResource's configuration profile by building and requesting a URL that matches the eResource's API presenting the searched content to directly to the end user.

Since the current KR structure offer a robust XML-oriented environment in terms of content validation and transformation, the Infobutton Manager was entirely implemented based on XML documents (for building eResources profiles) and XSLT for HTML/JavaScript for client-side presentation. The major advantage of this approach is the ability to update the eResources configuration and even the Infobutton Manager behavior itself with no downtime or recompilation.

Findings

KM Growth at Intermountain Healthcare

Figure 1 illustrates the growth of content in the knowledge repository from 2003 to the present. Additions of new content as well as revisions of existing content have grown steadily over time. The sharp peaks in the content numbers typically coincide with large batch imports to the repository. The first peak corresponds with a sizeable import of medication common list files from a medication ordering program; the second represents the loading of Intermountain's clinical element model library to the database. Collectively, the KR at Intermountain holds 31,642 unique documents. Counting all versions of these documents, the KR houses 111,986 distinct knowledge assets.

Table 1 illustrates monthly logins to KAT and KRO, Intermountain's authoring tool and content browser portal for the KR. Growth and usage have remained steady over time, with increases reflecting a broader set of engaged users in these platforms. Data in 2012 is current only until the end of Feburary.

Figure 2 depicts content delivery metrics for knowledge artifacts in the repository. Each time a user or application invokes a request for a knowledge artifact from the KR, it logs a request against our 'Document' service. On average, this service is invoked about 70,000 times daily, resulting in an overall volume of nearly 2 million hits per month.

Figure 3 depicts infobutton usage on a monthly basis at Intermountain Healthcare. Users have access to this resource from the problem list, medication ordering modules, and the lab review modules within HELP2, Intermountain's web-based medical record. Hits have increased from ~2800 per month in 2007 to over 16000 monthly in 2012.

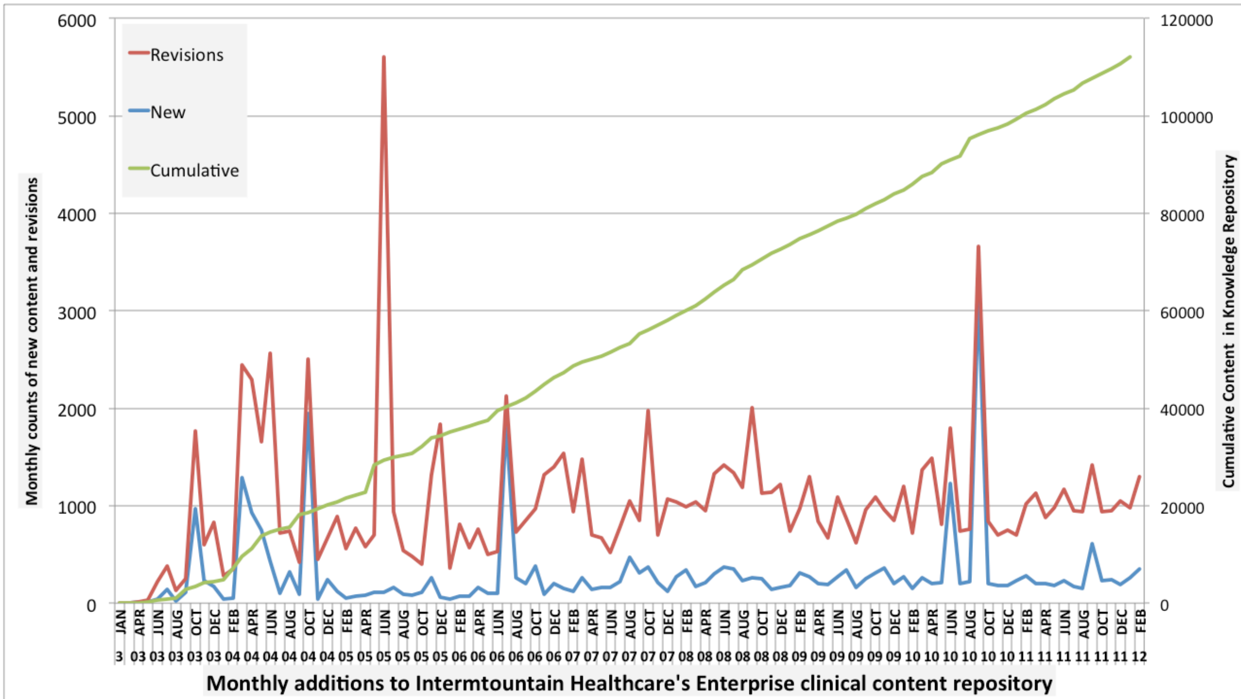

Figure 1. Content creation rate in Intermountain Healthcare's Enterprise Clinical Knowledge Repository (Jan 2003- Feb 2012). Monthly counts of 'new content' and revisions to existing KR content are shown. (primary vertical axis: left), cumulative content totals superimposed (secondary vertical axis: right)

YEAR	Unique KRO users	KRO Logins	Unique KAT Users	KAT logins
2004	272	6427	114	3427
2005	337	22647	137	22414
2006	376	25897	153	27035
2007	519	7195	139	5931
2008	724	10662	198	6693
2009	900	10417	267	5851
2010	1142	11751	238	4970
2011	1224	13165	248	5734
2012 (thru February)	519	2724	110	1166

Table 1 – KAT /KRO usage. Unique users per year and unique logins per year are shown.

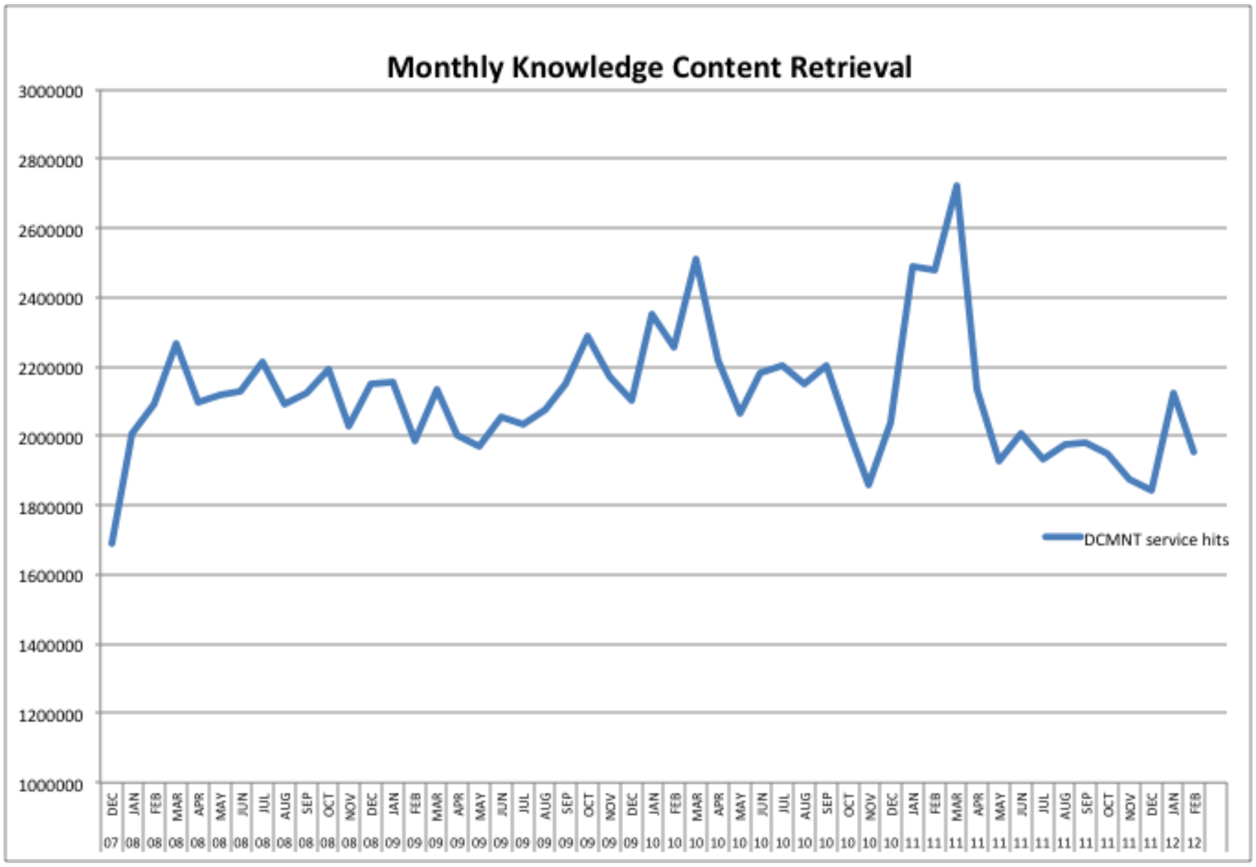

Figure 2. Monthly hits on the KR ‘Document’ service (which delivers content to users and applications)

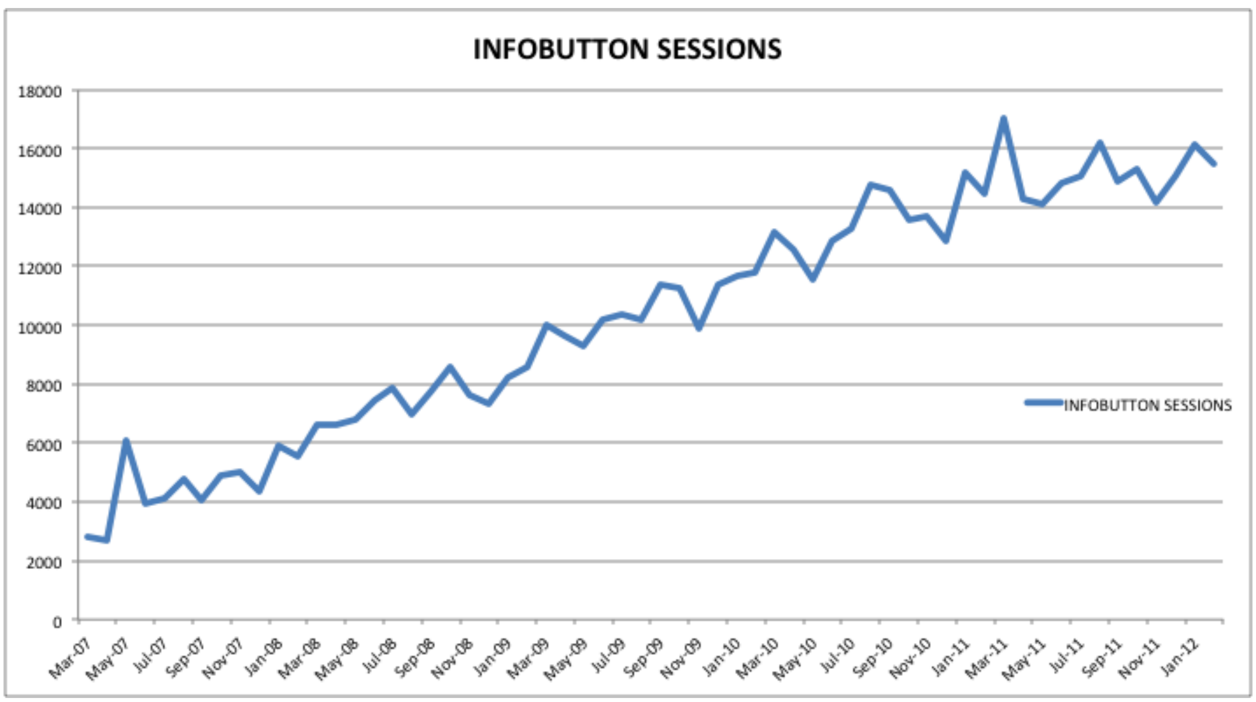

Figure 3. Monthly hits on the Infobutton Manager at Intermountain Healthcare

Discussion

Knowledge Management Needs

After more than 10 years of Knowledge Management at Intermountain it is clear that the evolution of KM processes locally are still in their early stages. Most of the focus of our development and content creation efforts has targeted processes like centralizing content, versioning, intelligent delivery, and high-level structured content capture. We feel that there is extensive work remaining needed to tool our services, applications, and users so as to be able to capture knowledge in ways that is fully computable and human readable. Our hope is to replace extensive narrative content with semantic-based content that is extensively structured and coded.

One of the powerful capabilities of the KM system at Intermountain is the ability to provide the users a user-friendly XML web-based authoring tool to build and publish knowledge assets. However, one of the problems that the current authoring tool has is the lack of advanced, synchronous form processing features during authoring time, owing in part to the outdated ActiveX components used in the authoring framework. Enhancing the tool to support these types of processing could help authors to create documents in a more productive way by executing validations and quick 'lookup-based decision functions during the editing process.

When the knowledge repository was designed, we supported two main delivery contexts. These set boundaries of content delivery internal or external to Intermountain's firewall. This type of constraints allows content managers to assign distribution rights in either or both of these domains. While this has served us well, we have more recently dealt with issues regarding content delivery via a third pathway: content requested the internet, but via an authenticated gateway (Intermountain's externally-available corporate intranet zone). At present, our content delivery service isn't connected to the necessary security framework that would allow content delivery via this externally-available authenticated gateway to our clients. We are working with our e-business team to develop this shared authentication token between our content delivery service and their external content portal.

The knowledge creation and maintenance workflow currently in place relies on the groups of users and Clinical Programs to make sure only pertinent materials are being produced and published. The current software structure does not provide a system workflow enforced environment where documents can be published only if they meet certain criteria and approvals signatures. Although we do have a workflow driven review toolset for systematic review of clinical content, this is not yet fully coupled with the processes that govern the creation and publication processes within KAT and KRO.

Key Themes of Current Knowledge Management Development

Our current strategy has served us well, but has also helped us identify areas for improvement and enhanced efficiency.. In an effort to address a series of development-focused and business needs, we have adopted a new strategic initiative that changes the direction of our efforts. The key themes we are trying to address are listed below.

a. *Improve sharing of common knowledge* – The redundancy of information in our KR potentially results in inconsistencies, which decreases the quality of care for patients. It also results in wasted time and effort for maintenance that could otherwise be better spent in clinical care.

b. *Inflexible representation of knowledge* – Too often, there is isomorphism between the representation of knowledge and the way it is being used. This mandates very frequent code changes. A more dynamic approach is needed.

c. *Improved performance* – Systems trying to alleviate the previous issue may end-up being purely based on dynamic transformations, which results in slow performance, not suitable to a real-time clinical environment.

d. *Usability* – Very often, the information is not readily consumable and first needs to be pre-processed. For example, a clinical rule may live in a repository system in some kind of source-code for clinicians to be able to modify, but in order to be used, it would have to be transformed into binary code. Often, systems lack support for multiple representations of the same knowledge asset, or are unable to enforce a formal process to keep these representations in sync.

e. *Componentized knowledge storage strategies* – The more monolithic the knowledge, the less it lends itself to reuse. There is a greater and greater need to be able to aggregate related pieces of knowledge.

f. *Enhanced Versioning* – Due to their need for KM systems to be fully 'traceable' for audits and other compliance-related analyses, we intend to enhance our versioning strategy, particularly as it relates to component interdependencies. The versioning capability is often either incomplete, or extremely space / time consuming, due to the monolithic nature of large chunks of knowledge.

g. *Backward-compatibility* – Not only do systems need to support new requirements, they also need to remain compatible with existing legacy applications, which tend to stay for a very long time in the medical field. This is often accomplished by brittle mechanisms, such as nightly transfers / migration from one system to another.

h. *Enhanced Authoring* – It is often difficult for someone modifying a knowledge asset to see the full ramifications of the change. This is due to the fact that dependencies and interactions between components are often not fully understood, due not only to the sheer size and complexity of the medical field, but also to a lack of formal dependency management on the system's end.

i. *Metadata authoring* – Given the universal, inflexible nature of our metadata model, medical experts sometimes have their hands tied as far as defining the blue-print and structure of some new type of information they need to publish. They have to rely on software developers to implement their request, and it often results into significant delays. As a result, many potential improvements that could have helped in patient care get either delayed or canceled altogether due to lack of resources. Therefore, there is an increasing need for systems able to let these experts create their own metadata.

j. *Component interdependency registries* – Our current infrastructure does little to support the understanding of how knowledge assets inter-relate with each other (either through semantic relationships or structural interdependencies). As we proceed with our next generation suite of tools, we intend to build robust support that allows users and systems to quickly understand content dependency trees.

Next-Generation KM features

In detailing our plans to address these issues, we have outlined below a series of features that will be incorporated in our next-generation KM framework. We anticipate that the lessons learned from our first decade of KM development will allow for richer use of the platform and lead to systems that are able to more intelligently benefit from the features listed below.

1. *Multiple representations of a same piece of knowledge: (e.g.*, Knowledge Asset), with exactly one of these representations being the source. The source can change over time as a better representation may become available, or as needs change. Any modification to the knowledge asset will be initiated on the source, triggering a reconciliation process on the other representations (either automated whenever applicable, or manually if necessary).

   Issues addressed:

   - *Allows users to pick-and-choose which representation is best suited for a specific task,*
   - *Provides support for pre-compiled / run-time ready representations,*
   - *Uses most rigorous and editable representation as the source,*
   - *Allows older format to 'linger' as one of the representation for support of legacy applications*

2. *Associative representation strategy, emphasizing highly reusable components:* This approach will both minimize space requirements and encourage re-use, as opposed to the monolithic knowledge representation strategy sometimes used today. This will also allow for ease of sharing of common components across multiple knowledge assets.

   Issues addressed:

   - *Emphasizes assembly of existing components whenever it applies*
   - *Greatly increase flexibility as far as knowledge representation*
   - *Breaking up knowledge assets into components,*
   - *Much more efficient and space-consuming versioning*

- *Potentially better visualization / ease of authoring – Allows user to concentrate on a specific component*
- *Associative models facilitate the metadata-driven approach.*

3. *Efficient versioning:* Encompasses any prior knowledge asset, simplified by the nature of the model – only the unchanged associations to existing components get copied (as "pointers" to child components), as opposed to making duplicate copies of the entire asset

   Issues addressed:

   - *Speeds up the versioning process (mostly copy associations, rather than full content)*
   - *Fully supports any previous versions (snapshots)*
   - *By always guaranteeing a faithful and complete snapshot it is possible to track any changes that have occurred, resulting in better trouble-shooting capabilities*

4. *Migration plan:* We plan to ease the process of of migration from existing / different representation systems, enabled by the multiple representation capability. For example, a previous knowledge asset might be represented as a large XML document. The new infrastructure will allow for a seamless migration of this XML to the new infrastructure, also as an XML document. If later, we decide that we want the source representation of this document to be composite (which this new architecture encourages, but does not mandate), then the composite representation will become the primary (source) representation, and the XML will be relegated to an alternate representation (which which still remain backward compatible with legacy applications if needed).

   Issues addressed: Backwards compatibility

5. *Real-time, "in-context" authoring* -- During the authoring process, a user will have the capability to browse a knowledge asset, drill to a component, and start making change to that component – This action will trigger the creation of a WIP (Work-in-progress) copy of the parent knowledge asset, in which only the root asset and the links to its unchanged components will be copied. Meanwhile, a new copy of the modified component will be created and linked to that new version of the parent, immediately putting the new component "in-context" within the parent asset. This will make it possible during authoring of the component, to see the effect of changing that component within the context of its parent and siblings.

   Issues addressed: Authoring richness and dependencies

6. *Meta-data driven infrastructure* (perhaps the most important aspect of this approach) – The structure and properties of a composite category will be for the most part represented as meta-data inside the database, making it possible to implement almost fully generic code. The flow of the processing logic (storing, retrieving and even rendering) will be essentially driven by that metadata, since the metadata will describe the nature of the associations between a parent-category and a child-category within a specific context. We envision that at some point, this will even enable "power-users" themselves to define the blueprint of a new category they would be responsible for. This could be done without having to write any new code.

   Issues addressed:

   - *Inflexible representations of knowledge metadata*
   - *Metadata authoring environment*

7. *Generic associations* – The model will go beyond the simple parent-child type of composition. The intent is to use metadata not just as a mean to dynamically describe composition, but as a blue-print to any type of semantic associations. The association metadata will also dictate whether a change to one end of the association should trigger versioning of the other end, and whether this versioning should be cascaded further.

   Issues addressed: Component interdependencies.

8. *Duplication checking*: We plan to implement efficient hash-key based lookups of component to prevent creation of duplicates. Hash-keys will be stored in the database during the creation of an asset, for quick look-up later on.

Issues addressed:

- *Performance Issues*
- *Redundancy mitigation / content integrity*

In early years of the project, many of the tools and services developed by the knowledge repository team served functions both in the spheres of knowledge management and enterprise document management. While we recognize the overlaps in these areas, we have opted specifically to focus more of our ongoing efforts in building infrastructure in support of knowledge management, realizing that some strictly 'document management' needs can and will be served by 3rd party products and vendors. We have chosen to focus more on tools and processes that capture semantic meaning in the representation of the knowledge itself. Our primary content targets revolve largely about content that is intended for consumption within runtime systems, not necessarily just document viewers.

Some of the above decisions are reflective of an increasing need to support knowledge management functions to other internal systems. Our terminology services and data modeling groups have creation tools but lack KM infrastructure. As such, a service-based focus with an increased emphasis on APIs and interactions with tools that create and modify knowledge content has realigned our priorities. While we still expect to provide and service authoring tools and content review/development applications, we also plan to support internal customers that bring their own creation tools to the table, but lack the intelligent middleware that provides robust KM functionality.

Conclusion

While much remains to be done in terms of future development at Intermountain Healthcare in our KM platform, we are pleased with progress and experience gained over the past 10 years. As we address the issues identified and build on successes achieved, we hope to create a system that continues to serve the enterprise by delivering clinical knowledge content for consumption in useful, timely, and intelligent ways. By emphasizing content re-use, semantic connections between content assets and flexible architectures, we anticipate being able to meet our clients' needs for knowledge delivery in powerful new ways.

References

1. Bali, RK. Clinical knowledge management: opportunities and challenges. Page iv. Idea Group Publishing (June 2005).
2. Fischer G, Ostwald J. Knowledge management: problems, promises, realities and challenges. IEEE Intell Syst. 2001;16:60–72.
3. Sittig DF, Wright A, et al. Comparison of clinical knowledge management capabilities of commercially-available and leading internally-developed electronic health records. BMC Med Inform Decis Mak. 2011 Feb 17;11:13.
4. Pryor TA, Gardner RM, Clayton RD, Warner HR. The HELP system. J Med Sys. 1983;7:87-101.
5. Gardner RM, Pryor TA, Warner HR. The HELP hospital information system: update 1998.Int J Med Inform. 1999 Jun;54(3):169-82. PMID: 10405877
6. Intermountain Healthcare. Available at www.intermountainhealthcare.org. Accessed on Mar 1, 2012.
7. Hulse NC, Rocha RA, Del Fiol G, Bradshaw RL, Hanna TP, Roemer LK. KAT: a flexible XML-based knowledge authoring environment. J Am Med Inform Assoc. 2005 Jul-Aug;12(4):418-30. Epub 2005 Mar 31. PMID: 15802477
8. Wilkinson SG. Knowledge repository online: an electronic knowledge development and maintenance process. Salt Lake City: Department of Medical Informatics, University of Utah, 2003, p. 156.
9. Del Fiol G, Rocha RA, Bradshaw RL, Hulse NC, Roemer LK. An XML model that enables the development of complex order sets by clinical experts. IEEE Trans Inf Technol Biomed. 2005 Jun;9(2):216-28. PMID: 16138538
10. Roemer LK, Borsato EP, Hulse NC. Glycemic control through computerized subcutaneous insulin calculators. Stud Health Technol Inform. 2009;146:473-7. PMID: 19592888
11. Hanna TP, Rocha RA, Hulse NC, Del Fiol G, Bradshaw RL, Roemer LK. Customized document validation to support a flexible XML-based knowledge management framework. AMIA Annu Symp Proc. 2005:291-5. PMID: 16779048
12. Del Fiol G, Haug PJ, Cimino JJ, Narus SP, Norlin C, Mitchell JA. Effectiveness of topic-specific infobuttons: a randomized controlled trial. J Am Med Inform Assoc. 2008 Nov-Dec;15(6):752-9. Epub 2008 Aug 28.
13. Cimino JJ, Elhanan G, Zeng Q. Supporting Infobuttons with terminological knowledge Proc AMIA Annu Fall Symp 1997:528-532.