IT University of Göteborg

CHALMERS | GÖTEBORG UNIVERSITY

Comparing Open Source and Proprietary Enterprise Content Management Systems:

Alfresco Compared to IBM Lotus Domino Document
Manager Integrated with IBM Lotus WorkFlow

DANIEL G.R. ANDERSSON

CHALMERS | UNIVERSITY OF GOTHENBURG

Abstract

This thesis is a part of the current discussions of open source versus proprietary software for Swedish authorities. A new governmental policy has been established for which the purpose is to spread the use of open source software and thereby save governmental funds and to increase quality.

A case study has been conducted about an enterprise content management solution for the authority the Swedish Armed Forces. Enterprise content management is not a new product category but an integrated approach to handle all types of content. The solution is required in order for the authority to conform to current Swedish laws. Two candidate systems are compared against the requirements for the solution, were one is open source and the other is proprietary.

The first research question addresses the argument of open source software lacking features. The result implies that open source software provides an equivalent set of features.

The second, third and fourth research questions address implications for Swedish authorities of choosing an open source or a propriety system.

The organizational implications are of democratic interest. Open source increases the organizational transparency by making source code public, which allows automated decisions to be examined. All citizens of Sweden are by law granted access to records that are not classified. Why should this not apply to the source code used by Swedish authorities as well?

Another possible organizational implication could originate from an unexpected multi-national scenario. It could become disastrous if Swedish authorities are dependent on abroad organizations which could not provide critical support. Skills and know-how should be supplied within the country.

The economic implications are profound. The Swedish authorities are currently purchasing solutions based on proprietary software from private companies. When the projects are finished, the companies are able to sell the same solutions again to other authorities for the same amount of money. Governmental funds could be much better spent if the authorities choose an open source solution and share the source code and experiences with each other.

Contents

Acknowledgements

Chapter 1

Introduction

Open source and proprietary (or closed source) software are two opposite approaches to the development, control and commercialization of software. In contrast to proprietary software, open source allows individuals and organizations to view, modify and redistribute the source code. During the commercialization of software through the 1970s and 1980s, software source code became managed as a trade secret. Only the object code was distributed and installed, the source code became hidden from the users. The idea was to gain an economic advantage over competitors and customers.[37] Today, is the information technology business switching back to open source. According to Gartner Research vice president Mark Driver, the better part of software will consist of open source code.

> You can try to avoid open source, but it's probably easier to get out of the information technology business altogether. By 2011, at least 80 % of commercial software will contain significant amounts of open source code. - Mark Driver at Gartner Open Source Summit 2007 [15]

1.1 Purpose

The relative merit of open source versus proprietary software has been widely debated in Sweden. News features are published more or less daily. In the area of enterprise content management, this thesis investigates implications for Swedish authorities when choosing open source or proprietary software. Technical, organizational and economic implications of choosing either will be discussed. A question often raised, is if open source can deliver the same features as proprietary software. For this thesis, a case study is carried through, comparing features of an open source enterprise content management system against a proprietary content management prototype.

1.2 Research questions

Based on the purpose, the following research questions were formulated.

- Does an open source content management system provide an equivalent set of features compared to a proprietary (closed source) counterpart?

- What are the technical implications for a Swedish authority when choosing an open source or proprietary content management system?

- What are the organizational implications for a Swedish authority when choosing an open source or proprietary content management system?

- What are the economic implications for a Swedish authority when choosing an open source or proprietary content management system?

1.3 Outline

The first chapter "Introduction" is an introduction to the thesis. Purpose, research questions and this outline are described to provide the readers with an understanding to their forthcoming reading.

The second chapter "Theory" introduces the theoretical framework for this thesis. The area of enterprise content management is a base for the reader to better understand the presented results. The open source description is mainly provided for the discussion chapter.

The third chapter "Research Method" describes the methods used in the thesis and a short description of the research process is presented. This is where the delimitation for the thesis is found, since some understanding of enterprise content management is required, found in the previous chapter.

The fourth chapter "Systems Evaluated" describes the software products that are compared in this thesis.

The fifth chapter "Requirements of the Swedish Armed Forces" presents the case study and the organizations and products that are included.

In the sixth chapter "Results" are the two systems features compared based on selected requirements in the pre-study for the Swedish Armed Forces' case and document management. In addition to this, is also comparison of general system and business concerns also included.

The seventh chapter "Discussion" is focused on discussing the research questions and the results, and an interview with the Swedish authority City of Stockholm's Culture Administration is included.

The eighth chapter "Conclusions" summarizes what conclusions that can be made from the thesis.

Chapter 2

Theory

The theoretical departure for the thesis is theories within the area of enterprise content management and open source. In this chapter are relevant theoretical literature presented and discussed.

2.1   Enterprise Software

The term enterprise refers to an organization of individuals working together to achieve a common goal. Organizations come in all shapes and sizes. But the term enterprise is often used for large, for-profit organizations.[1]

Enterprises generally have some common needs, such as managing information, assets, resources, customers, and so on. The term enterprise software is used to collectively refer to all software involved in supporting these common elements of an enterprise.[1]

2.2   Enterprise Content Management

Enterprise content management is a relatively recent term that is used to differentiate content management systems that can handle content with a broad definition. This type of system is able to handle all types of content, including business documents, web-based content, electronic transactions, e-mails, document images, and rich media. Enterprise content management is not a new product category, but an integrated approach. Document management, content management, enterprise portals, knowledge management, and collaboration tools are converged. There is an overlap between these areas which is reflected in the development of enterprise content management systems, and there is a need to take a coherent approach to the management of information within the enterprise. The core functionality of each of these areas is described further on in the thesis.[22] Smith & McKeen describes enterprise content management in a compact manner in the following quote.

> "Enterprise content management (ECM) is an integrated approach to managing all of an organization's information including paper documents, data, reports, web pages, and digital assets. ECM includes the strategies, tools, processes, and skills an organization

needs to manage its information assets over their life cycle." - Smith & McKeen 2003 Developments in Practice VIII: Enterprise Content Management[36]

Another definition is available from AIIM, a non-profit organization about managing content that was founded as early as 1943. This definition is very similar to the one above by Smith, but it also includes unstructured information. This formulation seems to make things clearer when explaining enterprise content management.

ECM is the technologies used to capture, manage, store, preserve, and deliver content and documents related to organizational processes. ECM tools and strategies allow the management of an organization's unstructured information, wherever that information exists. - AIIM website[2]

2.2.1 Document Management

The core of both systems in this thesis is document management features, and this is a good choice of area to begin for better understanding of enterprise content management. This is more or less where content management began. In the early days, it initially dealt with the interface between printed document and computer systems. Today electronic documents are an integral part of business processes and document management systems include indexing and retrieval, workflow capabilities, versioning, document check-in/check-out, collaboration and distribution. Today's systems are able to handle the relationship between documents and business processes, and are capable of managing the entire document life cycle. Features such as auditing, security and authorization, and document archiving, are as important as the content of the document itself.[22] This thesis will mainly focus on the Document Management capabilities when comparing Alfresco and the IBM based prototype.

2.2.2 Web Content Management

This area of enterprise content management primarily focuses on web-based content, but it is not necessarily limited to HTML and XML content. Whilst these formats are the most common storage and presentation formats for web pages, the overall site will also include many other content formats, such as images, audio, video, word-processing files, PDF files, and other specialized formats, such as software code. Web content management ranges from basic applications that are capable of handling a company intranet, up to full-blown systems that specialize in supporting the largest scale e-commerce initiatives, and the management of multiple Web properties, on a global scale. The fundamentals of web content management are the ability to separate content from its presentation, using templates to simplify the process of content creation and contribution, without the intervention of technical staff to publish material. High-level web content management systems will include the sophisticated workflow and administration tools that are required to manage complex web sites, and large numbers of content contributors. Some systems will also support mass personalization of web content, suitable for large-scale e-commerce initiatives in both consumer and business-to-business environments.[22]

2.2.3   Enterprise Portals

Portals act as a unifying layer that abstract the user from the underlying applications, and from the complexities of diverse data sources. It simplifies the process of information discovery in support of business decisions, and is often used as the basis for the creation of a knowledge network, enabling collaboration on a wide range of projects. The strength of a portal lies in its ability to draw together the information and applications, both internal and external, that are available to an organization, and present a personalized view to employees to add value to their everyday work.[22]

2.2.4   Knowledge Management

Knowledge management is a broad and open-ended area. But the main focus is that the tools are designed to maximize value from intellectual assets and making a direct connection between an enterprise's intellectual assets and positive business results. In practice, knowledge management often includes identifying and mapping intellectual assets within the organization, generating new knowledge for competitive advantage within the organization, making vast amounts of enterprise information accessible, sharing of best practices, and technology that enables all of the above, such as groupware and intranets.[11]

2.2.5   Records Management

This area is similar to document management and is a part of the enterprise content management area. But it can be interpreted as addressing an electronic system for managing paper records or for managing electronic records. The electronic system requires a server and the other physical paper system would rather require a building. One possible definition of an electronic records management system could be as follows.

> An automated system used to manage the creation, use, maintenance and disposal of electronically created records for the purposes of providing evidence of business activities. These systems maintain appropriate contextual information (meta data) and links between records to support their value as evidence. Electronic Records Management System (ERMS) are a subset of business information systems whose primary purpose is the capture and management of digital records. - National Archives of Australia[28]

This definition indicates that this area is not about managing physical records. However, there are records management systems that provide features to handle physical records.[24]

2.3   Open Source Versus Proprietary Software

This section provides a foundation to the Discussion chapter.

Open source software is growing in popularity in both the commercial sphere and the number individual end-users are increasing. The open source development methodology has been heralded by some of its proponents, such as Eric

Raymond[34], as a superior way of producing software code. Whether open source software is indeed faster, better, and cheaper is a matter of controversy.

To begin with, what is open source? Roughly, it means that the source code is made public, and that the modifications made by its users also is turned back to the community. The details vary with the license adopted for the software. Some of the key criteria included in the Open Source Definition[30] are (a) the royalty free redistribution of the program, (b) the release of the source code, and (c) the requirement that all modifications be distributed under the same terms as the license of the original software. Open source software should not be confused with shareware (which is freely distributed, but whose source code remains proprietary) and public domain software (which is not licensed and thus available to anyone without constraint).[25]

The today common, proprietary software (or closed source) development methodology means that customers pay for a nonexclusive license which allows to them to use the software, but not to view or modify the source code. In other words is the software not sold as it is commonly formulated, it is rather leased to the customers. Customers are restricted in their use, modification and copying using technical or legal means and often both. Technical means can be to only provide machine-readable binaries and withholding the human-readable source code. Legal means can involve software licensing, copyright, and patent law.[39]

The open source development methodology, with Eric Raymond at the front, is about offering practical accessibility to software source code. There is also Free Software Foundation's with Richard Stallman at the front which as a somewhat different view on the matter. The Free Software Foundation also propagates the practical benefits of open source, but they wish to emphasize the ideological aspect of freedom. They in particular wish to emphasize liberty and prefer the term of free software instead of open source software to stress this. The Free Software Foundation view free software as a matter of the users' freedom to run, copy, distribute, study and modify software.[37]

A software license is a contract between the software publisher and the user of the software. This is where the difference between open source, free, and proprietary software is defined. An example of an open source license is the BSD license. It more or less allows anyone to do anything with the source code, with the exception three paragraphs about copyright notice and promoting derived products.[27] Any software using this license can become proprietary and only the object code will be distributed without the source code. This is by many considered as maximum flexibility and freedom, but the Free Software Foundation wish to prevent this from happening. The Free Software Foundation wishes to always make sure that free software stay free. Any software in the public domain is also possible to make proprietary in the same way as the BSD license. To avoid free software of becoming proprietary, the Free Software Foundations have created their own software license. The Gnu's Not UNIX's General Public License, or more commonly known as GNU GPL or just GPL, is a strong copyleft license for software and other kinds of works. Copyleft is a general method for making a program or other work free, and it requires that all modified and extended versions of the program to be free as well. This license was originally written by Richard Stallman for the GNU project and it is used for example by the Linux kernel.[37]

2.3.1 The Current Discussions

There are and have been a lot of discussions of open source versus proprietary software and Swedish authorities. In a news feature at Computer Sweden the 12th of May earlier this year, it is written about a new policy that proclaims that open source software is to be evaluated with the same criteria as proprietary software. The most affordable software is to be prioritized. It is intended for internal use at Verva when they define future framework contracts. Verva is the organization that coordinates the development of central government and the procurement of framework contracts concerning information technology products and services, for the entire Swedish public sector. The purpose of the new policy is to spread the use of open source software and thereby save taxpayers' money and to increase the quality in the governmental systems. The branch head of the department of development support at Verva, Jenny Birkestad, states that open source increase the reuse of software in the public administration. She points out that many authorities have similar information technology requirements and that reusing software would be of great benefit. Verva believes that open source supports commercial competition and Verva stress the importance of escaping vendor lock-in and dependence on single vendors. According to Jenny Birkestad, it is a beneficial side effect that the source code can be examined since this makes it possible to review how automated decisions are taken. She state that this makes open source a question of organizational transparency and openness.[23] The case study for this thesis is a part of this discussion, and the upcoming open source alternative investigated is the popular[12, 26] enterprise content management system Alfresco.

Chapter 3

Research Method

This chapter describes the method of the thesis. This is to ensure the possibility for someone not included in the original research process to examine the investigation.[29]

3.1 The Research Process

The research process includes identifying the problem area and formulating the purpose and research questions, literature studies, possibly specifying the problem area further, choice of method, choice of inquiry group, how the empirical data was collected, course of action, processing and analyzing, and reporting. This is an ideal view of the research phases in a logical order, but in reality the phases might overlap each other. New knowledge is also unraveled during the research process that can be added. This makes the research process iterative.[29]

3.2 Scientific Viewpoint

This thesis does not aim to confirm or to disapprove a theory, but to investigate a specific area which results in an inductive approach. The information acquired in this thesis is descriptive, since it is not juxtaposed against a hypothesis. A social constructivism aspect is applied, where the research approach focuses on meanings and gaining understanding of a phenomenon of using conversation. In social constructivism, it is not possible for the researcher to be objective. The researcher is a part of the process by the act of gathering information. In an interview situation for example should always the aspect of the interviewer's influence on the interviewee and the results as well as the interpretation of the results be taken into account when the conclusions are assessed.[29]

3.3 Choice of Method

To gain an in-depth understanding of a phenomenon and understanding a specific organization, is a qualitative research approach the most appropriate. One form of collecting data for a qualitative research study is through interviews,

which was the primary form of empirical data for this thesis. It is most flexible Focus was on a smaller selection of interviewees for in-depth data. Qualitative research methods are flexible, dynamic, and variations in the material are allowed. The disadvantages with a qualitative method are its time-consuming nature and it may be difficult to analyze and understand all of the aspects in the empirical data.[29]

A comparative method was used for the case study. Differences and similarities between the two systems are described in the result chapter.

3.4 Choice of Inquiry Group

The case study investigates a project at the Swedish Armed Forces. This Swedish authority has slim to none automatic support for providing public access to public records. This makes it difficult for this authority to conform to current Swedish laws. This resulted in a major project at the consulting firm Atea which has conducted a pre-study. If this pre-study and other empirical data is utilized properly in a thesis such as this, valuable scientific insights might arise about a Swedish authorities and information technology. In this case study the subject of open source and proprietary software and Swedish authorities is discussed.

To further establish and generalize the discussion of Swedish authorities and open source or proprietary software, an interview were conducted with the Swedish authority City of Stockholm's Culture Administration.

Employees at Atea were interviewed to obtain empirical data about the IBM based prototype. The company is the paramount IBM provider to the public sector.

To obtain empirical data about Alfresco, were interviews conducted with employees at Alfresco and at a consulting firm which is partner to Alfresco.

3.5 Delimitation

This study is based on a case, with the Swedish Armed Forces. The requirements of this case study are used to limit the comparison. These requirements include a partial set of Alfresco features which is an enterprise content management solution. Therefore, some features such as managing web content such as Internet sites and intranet sites, indexing, managing digital imaging, scripting, social computing and templating will not be addressed. To avoiding the result from becoming bias in any direction, the selected features are then generalized headings originating from the enterprise content management area, stated in the second definition from AIIM in the theory chapter "ECM is the technologies used to capture, manage, store, preserve, and deliver content and documents related to organizational processes." In addition to this, are also general system and business concerns addressed.

3.6 Course of Action

Interviews

Unstructured interviews are similar to conversations that focus on a particular topic and may often go into considerable depth. Questions posed by the interviewer are not predetermined in the format and content of answers. The interviewee is free to answer as fully or as briefly as she wishes.[33]

Semi-structured interviews were conducted as they combine features of structured and unstructured interviews and use both questions that require a precise answer and questions that the format and content of answers is not predetermined.[33]

On March 11 2008, a semi-structured interview was carried through with Alfresco during a meeting between Alfresco and Atea. David Vogel sale responsible for Scandinavia, together with a system engineer, were the interview subjects. Chief technical officer, Magnus Ericsson, was present from the information management department at Atea. This took place before my bachelor thesis course had begun.

On March 18 2008, the project manager at Atea for the Swedish Armed Forces project, Carl-Fredrik, was interviewed about the IBM based prototype as well as the about the Swedish Armed Forces organization and requirements. He is a former employee of 11 years at the Swedish Armed Forces and he possesses valuable insights of the authority. This was an unstructured type of interview, and most questions arise during the session. This took place before my bachelor thesis course had begun.

On April 7 and 8 2008, semi-structured interviews was carried through with an Alfresco certified instructor, Andrew Savory, from Sourcesense London.

On May 5 2008, an unstructured interview was carried through with system developer, Jonas Persson, at Atea. He is currently working with IBM Lotus software and was included in the project group for developing the IBM based prototype for the Swedish Armed Forces.

On May 7 2008, an unstructured interview was carried through with archivist Peter Foberg at the City of Stockholm's Culture Administration.

On May 23 2008, a follow up interview with Carl-Fredrik was conducted to clarify some of the empirical data.

Personal Communication

The thesis was written at Atea's office in Göteborg and during the spring of 2008 there was several short discussions with the chief technical officer at Atea's information management department to elicit technical information mainly about the IBM based prototype. The discussions were too short and too simple to be labeled as interviews.

Literature Survey

The information describing enterprise content management and its underlying areas, searching literature was the primary information source. Most of the information in this area came from articles and some books.

A legal document created by Atea and the Swedish Armed Forces is used as empiric data. It accounts for current the Swedish legal framework and the

requirements that it places on a case and document management solution for the Swedish Armed Forces.

The technology addressed in the thesis, specifically Alfresco, is on the cutting edge of enterprise content management. Useful articles concerning enterprise content management were found, but none comparing open source and proprietary systems.

Training

In order to gain a technical understanding of Alfresco, a system administration course for Alfresco was undertaken in London for two days. It was provided by the open source consulting firm Sourcesense.

3.7   Processing and Analyzing

Questions were prepared in advance for each interview, some spanning up to 46 questions. Interview notes were taken for each interview. The information for each heading in the result chapter is to a large extent based on the interview notes. It was time consuming to process and analyze marketing information of the two systems, into impartial information.

Chapter 4

Systems Evaluated

This chapter describes the software products that are compared in this thesis.

4.1 IBM Lotus

Lotus was founded in 1982 and IBM purchased the company in 1995. Today is IBM Lotus a set of proprietary software for supply application design and development; dashboards and business solutions; e-mail, calendaring and collaborative applications; instant messaging and web conferencing; applications to mobile and wireless devices and clients; social software; team collaboration, content management and e-forms. The back-end is made up of LotusScript, which is similar to Visual Basic but specific for only IBM Lotus software. JavaScript is often used for web development, and it is a common programming language.

IBM Lotus Domino Document Manager

IBM Lotus Domino Document Manager, formerly named Lotus Domino.Doc, is a proprietary product to organize documents for shared access by work teams. The software automates document processes like review and approval, versioning, publishing and archival, check-in and check-out features. The application programming interface is buggy and Atea do not use it to avoid problems. They alter the software source code directly instead which means they have to redo the modifications at each update. The product is poorly implemented and IBM is considering redoing the whole product from scratch and release a new version as open source. If there is any interest in the product it will continue to be developed by those who wish to, or it will simply stay in a maintenance mode.[17]

IBM Lotus WorkFlow

IBM Lotus WorkFlow is a platform for automating, refining and managing complex people-based business processes. It is out-of-the-box prepared to extend the native workflow capabilities of Lotus Domino software.

4.2 Alfresco

Alfresco Software was founded in 2005 by John Newton, co-founder of Documentum and John Powell, former chief operating officer of Business Objects. Its investors include the leading investment firms Accel Partners, Mayfield Fund and SAP Ventures. Alfresco claim to be the leading open source alternative for enterprise content management. They aim to couple the innovation of open source with the stability of a true enterprise-class platform.[6]

Alfresco, the product

Alfresco is made up of what Alfresco labels as the best-of-breed open source software. Java is the only programming language that is used which, according to the TIOBE Programming Community Index[13], is the main programming language of today. Installation is made easier with a single Java war file. It is 40 MB large and can be run in many different application servers. It can share the same Java Virtual Machine as an embedding application or be accessed remotely.[8] This makes Alfresco very flexible and it should be possible to run on more less all major operating systems.

The newcomer Alfresco has big ambitions and aim to surpass Documentum and Microsoft SharePoint in terms of features, functionality and benefits to the user community.[6]

Alfresco Community Edition

This is a free version available with the GNU GPL license. As for all other GPLed software, it is available "as is" and without any warranty. Anyone is able to download and test it. There is a large forum with tens of thousands posts. There is also a comprehensive bug handling system.

Alfresco Enterprise Edition

Exclusively offered to paying subscribers with a commercial license, similar to how MySQL, Red Hat, and other leading open source companies license their technology. Alfresco claims the license to be one hundred percent open source, but the Open Source Definition is not mentioned.[3, 5] It should probably rather be labeled as shared-source[18].

Chapter 5

Requirements of the Swedish Armed Forces

This chapter describes the case study of this thesis.

A case study has been conducted about a content management solution for the Swedish Armed Forces. The solution is required in order for the authority to conform to current Swedish laws, specifically the Swedish Principle of Public Access. The Swedish version of the law allows all citizens of Sweden access to all records that are not classified, and it is one of the strongest of its kind in the world. Laws changes with its society and at the present time, none of the hundreds of Swedish authorities probably conforms to them completely. The Swedish Armed Forces has slim to none automatic support for providing public access to public records. This result in deficient standards for naming documents, versioning, authorization control and so forth. The fact that there are few requests of any of the records at the Swedish Armed Forces is a major explanation to this.[16]

A pre-study has been conducted by the information management department at Atea to formulate the requirements for a content management solution for the Swedish Armed Forces to conform to Swedish laws. It was an extensive legal work and the document sums up to 49 pages and 90 requirements. The chapter headings are the Swedish principle of Public Access, public records, registering public records, administration of public records, archiving, consignment of public records, security related requirements and internal rules for the Swedish Armed Forces and Swedish Personal Data Act.[20]

Atea has operations in the 26 largest cities of Sweden and is a leading provider of infrastructure solutions. The key competences are communication, system, security, information management, IT-Infrastructure, IT-integration, procurement, product supplies, eSHOP, software licensing and management, home PC, print & supplies, security, consolidation and service of Hardware and Software. The private company is Sweden's largest communications supplier including at the Verva network agreement (public frame agreement), and the second largest provider on HP high-end storage solutions.[10] For this thesis, only the information management department at Atea is involved.

Following this study, a prototype was created by Atea. In Swedish it is la-

beled as "Försvarsmaktens Dokument och Ärendehantering (FM DÄH)", which roughly translates to "The Swedish Armed Forces' Case and Document Management Solution". It could also be labeled as records management solution. But as described in the theory chapter, is the records management area somewhat ambiguous and it could misinterpreted that this solution manages physical records. This is not the case and therefore will this term be avoided throughout the thesis. The solution in question is constituted by the proprietary IBM Lotus Domino Document Manager and the proprietary IBM Lotus WorkFlow which extends the native workflow capabilities of the IBM Lotus Domino Document Manager.[17] This case study compares this prototype against a possible solution using Alfresco.

This solution is meant to make use of cases. The cases can include internal working drafts and public records. The internal working drafts have to be approved as public records before they are available to the public, and could be meeting notes or any other file. The public records could be received by an administrator and then added to the corresponding case. Public records could consist of a recorded phone call, a word-processor file, a movie clip, pictures, spreadsheet or something else.[16]

Chapter 6

Results

This chapter compares features from selected requirements in the pre-study for the Swedish Armed Forces' case and document management. In addition to this, is also comparison of general system and business concerns also included.

6.1 Reliability

This is a general system concern. Down-time of a system could be very expansive or even disastrous in if manages critical functions.

This area has been well addressed in both systems, and they offer clustering and load balancing for high availability. Both systems are also highly scalable for future expansion.

One potential storm cloud for Alfresco could be start-up challenges. The company has only existed since 2005. But the management of Alfresco is experienced and has reference from among others Documentum, which is a top-end ECM solution. Alfresco has been used for solutions to support customers such as European Court of Justice and United States Department of Homeland Security. There are no known limitations in amount of storage or users for Alfresco, according to David Vogel at Alfresco, the limitations are the used database and other underlying infrastructure.[32, 38]

IBM Lotus has been around for many years and they are stable products. IBM Lotus WorkFlow is out-of-the-box prepared to be integrated with IBM Lotus Domino Document Manager, and Atea has successfully delivered several solutions using both products together.[17] One exception though, is the IBM Lotus Domino IBM Lotus Domino Desktop Enabler which allows users to access the repository as regular Microsoft Windows shared drives. This is an essential feature for the IBM based prototype, and unfortunately it is unstable and IBM is not developing it further. It is proprietary and therefore is a consulting firm such as Atea, not able to modify it. There is no alternative to this piece of software when using IBM Lotus Domino.[31]

6.2 Authorization Control

This area is included in AIIM's definition of enterprise content management as the technology to manage and deliver content. This is always a key issue for any system managing information.

Both systems provide equal features for authorization control such and groups and roles, inheritance of authorization settings from parents etc. The exception is that only Alfresco is able to control authorization for single files while IBM Lotus Domino Document Manager only handles folders.[16, 35]

6.3 Case Management

This area is included in AIIM's definition of enterprise content management as the technology to manage and deliver content. This is specific approach of managing content for the Swedish Armed Forces' case and document management solution.

A case corresponds to a space in Alfresco, which is a hierarchical folder structure. All containment (rooms, zones, folders) is managed through spaces.[4] It is possible to control access to single documents. This would allow that all documents of a case are classified but the meta data of that space is still kept available. This would allow the Swedish Armed Forces to conform to Swedish laws.[16, 35]

The IBM based prototype makes use of IBM Lotus Domino Document Manager's predetermined concepts in a hierarchically descending order of library, file cabinet, binder and document. In this definition is the binder only allowed to can contain one single document. This makes it somewhat problematic to create an optimal solution for the Swedish Armed Forces.[16]

6.4 Document Management

This area is included in AIIM's definition of enterprise content management as the technology to manage and deliver content.

The Swedish Armed forces will mainly use the content management solution for traditional document management. Therefore, features such as categorization, versioning, setting document types, share comments for documents, save attachments into the system from Microsoft Outlook, automatically sort documents (rules and workflow), working and synchronizing with the system using Microsoft Windows shares (SMB/CIFS), are important. The mentioned features are provided by both systems.[16, 35]

Alfresco does though have a limitation in the versioning functionality. The version identification is set to always start at "1.0". The IBM based prototype is based on IBM Lotus Domino Document Manager, for which it is possible to choose custom version identifications. This might be problematic for the Swedish Armed Forces and other large organizations.[35]

The two systems are providing comparable features concerning workflow. The difference is that Alfresco allows graphical modification workflows using

jBoss jBPM Designer, while manual programming is required for IBM Lotus
Domino Document Manager.[35]

6.5 Repository

*This area is included in AIIM's definition of enterprise content management as
the technology to deliver and store content.*

Both systems support the "JSR-170 Content Repository for Java technology
API", or as it is often named "Content Repository API for Java" (JCR). It
is a standard API to access content repositories in Java 2 independently of
implementation.[3, 14]

Alfresco make use of Hibernate which is an object-relational mapping library
for the Java language, providing a framework for mapping an object-oriented
domain model to a traditional relational database. Hibernate generates the
SQL calls and relieves the developer from manual result set handling and object
conversion, keeping the application portable to all SQL databases. This includes
DB2, Oracle, Microsoft SQL Server, MySQL, PostgreSQL and several more.

IBM Lotus Domino Document Manager is able to use IBM Lotus Domino
databases, DB2, Oracle, Microsoft SQL server, FileNet and more.

6.6 Customizing

*This is a general system concern. It might not be possible to properly implement
a system if it is not customizable enough.*

Alfresco is open source and all code can be modified. There are several files in
plain text format for customizing the system without modifying the source code.
They are quite straight forward to understand and there are useful comments
through the files.[35] David Vogel at Alfresco states that it is appreciated if
modifications are communicated to Alfresco, for possible inclusion in future
official releases of Alfresco.[38]

System developer Jonas Persson, currently working with IBM Lotus prod-
ucts at Atea, states that most of IBM Lotus Domino Document Manager is
available to the developer through IBM Lotus Domino Designer. Therefore, in
a perspective of open source or proprietary, it does not make a big difference
to work with IBM Lotus Domino Document Manager or Alfresco. The excep-
tions are IBM Lotus Domino Desktop Enabler, IBM Lotus WorkFlow Viewer
and IBM Lotus WorkFlow Architect which are pre-compiled. Other code is
available for IBM providers to modify and compile themselves.[31]

6.7 Extensibility

This is a general system concern.

For Alfresco, it is possible to create installable extensions. These are called
modules and are packaged as an AMP files (Alfresco Module Package). They
consists of XML, images, CSS, etc. that collectively extend the functionality

or data provided by the standard Alfresco repository. They could contain as a set of custom templates, a new category, a custom model and associated user interface customizations, or a complete new set of functionality. AMP files can be installed into the Alfresco WAR using the Module Management Tool. An AMP file has a standard format which can be customized if required.[16, 35]

IBM Lotus Domino saves a large part of customizations in IBM Lotus Domino sub-forms. These can provide customizations, but there is no feature to handle extensions.[17]

6.8 Updating

This is a general system concern.

In Alfresco, there are two sets of configuration and customization files in plain text format. They are stored in two separate folders. One of the set is independent of the version currently installed and its files can easily be copied to another installation. The other set contains configuration and customization files specific for the version currently installed. Any modification done to these files are not guaranteed to be compatible with future versions. Any contingent manual configurations and customizations made to these files, probably needs to be documented and be redone in the next installation of Alfresco.[35]

Everything that is meant to be modified, is modified using IBM Lotus Domino sub-forms. The sub-forms are included in the IBM Lotus Domino databases which are easily exported and imported.[17, 31]

The integration between IBM Lotus Domino Document Manager and IBM Lotus WorkFlow is standard, but it still needs some manual work when updating of the products.[16]

With a couple of mouse clicks, it is possible to export the whole repository in Alfresco into six .acp files. These are then as imported in another installation. These included everything expect indexes, which are easily rebuilt.[35]

In IBM Lotus Domino Document Manager stores all data in IBM Lotus Domino databases. These included everything expect indexes, which are easily rebuilt.[16, 35]

6.9 Releases

This is a general system concern.

The release cycles of Alfresco are tight and certified major release is delivered roughly once a year. It could be recommended to consider which new features that are included in future releases. The features presented in this thesis are provided in Alfresco Community Edition 2.9B and version 3.0 is to be released already in a couple of months after this thesis is written. Each year, Alfresco chooses five key areas to focus on. The key areas for 2008 [9] are as follows.

- New and enhanced team and enterprise collaboration services

- Publishing and management services for dynamic, Web 2.0-enabled websites

- Web client usability

- Scalability, performance, and enterprise-readiness

- Standards

But the situation is totally different for IBM Lotus Domino Document Manager. No more future releases are planned and the product will go into a maintenance mode. This is the main reason why Atea is investigating how to replace the current IBM based prototype for the Swedish Armed Forces. IBM will possibly rewrite the software and release it as open source, and let anyone who wishes to continue developing it.[17]

6.10   Partnership for providers

This is a general business concern.

Alfresco is concerned about growing too fast, which is a more conceivable risk as an open source company. Therefore, to enter a partnership with Alfresco a one-time fee of 10000 € is demanded and a minimum of two employees have to undergo certification training for Alfresco. This is also to make sure that the partner is committed and that they believe in Alfresco's product. It these requirements are not met, they do not hesitate to turn down any company no matter how well renowned they are.[38]

IBM does not place any similar requirements for a company to become a partner with them. Atea has a far-reaching partnership with IBM. But IBM is one of the largest companies in the world, and there is no organization in Sweden that fits IBM definition of "large". This makes it difficult to get IBM's attention for needed new features and such.[17]

6.11   Cost Estimation for Customers

This is a general business concern. License fees are a small part of the development total costs, but it is the only numbers available in the empirical data for this thesis.

Alfresco does not demand a one-time purchase fee. The standard support level, Gold, costs 16000 € per year.[38]

IBM Lotus Domino Document Manager and IBM Lotus WorkFlow can be obtained in a single software suite. The one-time purchase cost is 16000 €. There is an annual maintenance fee of 20 % to IBM of the 16000 €, which is 3200 € per year. Support costs are added in addition to this.[17]

6.12 Supported Interfaces

This area is included in AIIM's definition of enterprise content management as the technology to capture and deliver content.

There are several supported interfaces available in both systems. But for the Swedish Armed Forces will probably not FTP, REST, WebDAV and Web Services be of much relevance, these are included to help answering the thesis fourth research question.

	Alfresco [7, 21]	IBM based prototype [17]
Standard web access.	✓	✓
Microsoft Office 2003/2007 plug-in providing: check in & check out and save/open dialog with meta data fields	✓	✓
FTP: File transfer protocol to manage files on remote servers.	✓	✓, custom solution possible
WebDAV: "Web-based Distributed Authoring and Versioning" is a set of extensions to the HTTP protocol which allows users to collaboratively edit and manage files on remote web servers. It could be considered as the next step of evolution of FTP. The Microsoft Office plug-in makes use of WebDAV.	✓	✓, custom solution probably possible
CIFS/SMB: Shared access to files, printers, serial ports, and miscellaneous communications between nodes on a network.	✓	✓, local installation of a Desktop Enabler is required
REST: "Representational State Transfer" is a set of rules that an architecture should conform to.	✓	✓, custom solution possible
Web Services: a system designed to support interoperable Machine to Machine interaction over a network.	✓	✓, custom solution possible
JSR-168 Portlet Specification: interoperability between portlets and portals, this specification will define a set of application programming interfaces for portal computing addressing the areas of aggregation, personalization, presentation and security.	✓	✓

Chapter 7

Discussion

7.1 Features

The result chapter shows that the open source software in the case study provides an equivalent set of features compared to the proprietary counterpart. There is no reason to argue against open source software that it lacks important features because of the chosen development methodology.

7.2 Technical Implications

Open source makes it possible for consulting firms to modify software used for various solutions. This can be very useful. A customer might have very specific requirements, or no suitable off-the-shelf software may exist and comprehensive customizations cannot be avoided. This could also prove very useful if the product no longer is supported and updated, since the consulting firm itself could make the necessary modifications. In the case study, the Desktop Enabler used in the IBM based prototype is not developed further[31] and it is unstable. It would valuable to the consulting firm in this thesis to be able to improve the software, but they are unable to do so. The software is only provided in object code without the source code.

Open source software offers the possibility of a deeper understanding of the system and its operation because the inner workings are exposed to the user. This could be vital to the development of stable and reliable extensions to the software.

According to an archivist at City of Stockholm's Culture Administration, Peter Foberg[19], there exists a proprietary software vendor supply the full source code of their software. In spite of this, does the company consider themselves as proprietary software vendor.

Security is not addressed in this thesis, but it is a subject under discussion whether open source is more or less secure than proprietary software. The discussion is that additional vulnerabilities are fixed in open source solutions but at the same time is the system inner workings exposed to possibly malice individuals. This is an essential area for further research.

7.3   Organizational Implications

In the feature mentioned in the Theory chapter the branch head of the department of development support at Verva, Jenny Birkestad, states that it is a beneficial side effect that the source code can be examined since this makes it possible to review how automated decisions are taken. This could be of interest for guaranteeing personal integrity. Why should not concerned persons be able to confirm that software conforms to the Swedish Personal Data Act? One other aspect of a similar kind is that authorities are able to examine source code to guarantee that no malicious functions are included. When reasoning in risk management terms, one possible scenario could be that functions are included for espionage. How would the authority know if software is distributed only in object code?

The archivist at City of Stockholm's Culture Administration, Peter Foberg[19], gives the impression that his organization is not affected if open source or proprietary software is used. The critical factors are that necessary features are provided and that there is a trustworthy provider taking responsibility for debugging, support and future development. If there is a provider taking care of these factors, would probably the City of Stockholm's Culture Administration be satisfied with using open source.

The gained organizational transparency that open source code provides is of democratic interest, in particular for authorities. The Swedish Principle of Public Access allows all citizens of Sweden access to all records that are not classified. This is the very reason for the case and document management solution for the Swedish Armed Forces is to be realized. Why should this not apply to the source code used by Swedish authorities as well? Peter also states that when it comes to organizational transparency, it is more important if an authority take the cost to publish management routines for information, rather than the source code of the used software.

The archivist at City of Stockholm's Culture Administration, Peter Foberg[19] points out that well-designed software allows end-users to make small modifications with a user-friendly graphical interface. And if the source code is available, will most end users not possess the needed skills to implement the required modifications. Peter also points outs that his organization avoids to modify any code of the off-the-shelf software they use, to be able to guarantee stability.

One possible difficulty with open source is that the needs of its users is not necessarily prioritized, the developers might instead focus on what interests them. In the case study of this thesis, would Alfresco act as a middleman and solve this for its customers.

When reasoning in risk management terms, authorities should avoid being dependent on organizations abroad. It could prove disastrous in times of war, or other multi-national scenarios, when critical support cannot be provided. In the case study, both systems are provided by an abroad organization. But if the source code is available to the authority, it would be possible to support and develop the software if the competence is available nationally. These risks especially apply to the authority in the case study, the Swedish Armed Forces. What if Sweden would be at war with the country that the company supporting the software is located in? The other country would most surely find it to be out of the question to support a military organization in the country they are making war against.

7.4 Economic Implications

Although it might be a negligibly small part of a large development project, a generalization can probably be made that open source software has a lower, or no cost, of obtaining it. This is true for this thesis case study. But it is fallacious to assume that the total cost of ownership for open source software always is lower than proprietary counterparts. There are many dimensions to consider, such as implementation, converting information from other products, customization, support, maintenance and much more. The cost of a transition period including loss in productiveness, training costs, etc. is not determined by if open source or proprietary software is chosen. It is of greater importance how much the two solutions have in common. Low or the absence of license fees is a common argument for open source software. But one has to recognize that the purchase cost only is a part of the total cost of ownership. For example could a software vendor provide their software as open or free source, and then make sure that they themselves are the only ones that are able to integrate the software with other solutions and charge heftily for the consulting services.

Another more important reason for authorities to choose open source software is the collaboration possibilities. Currently are Swedish authorities purchasing solutions based on proprietary software from private companies. When the projects are finished, the companies are able to sell the same solutions again to other authorities for the same amount of money. Governmental funds could be much better spent if the authorities choose an open source solution and share the source code and experiences with each other. A recent initiative for this is Programverket. It is a project about open software in the public sector and the purpose is to achieve more collaboration and more efficient use of information technology within the public sector. According to an archivist at City of Stockholm's Culture Administration, Peter Foberg[19], no such collaborations are taking place today at his organization. The authority only use proprietary software and the main reason to switch to open source software would be cost savings.

Chapter 8

Conclusions

Open source software provides an equivalent set of features compared to proprietary counterparts. There is no reason to argue against open source software that it lacks important features because of the chosen development methodology.

Open source provides the advantage to be able to modify software as needed, and the dependency on the software producers is lowered. A deeper understanding of software is possible if the source code is available. This could prove vital. It is easier to evaluate software if it is open source. This saves the customer from time-consuming administration and the customer is also able to actually tryout the software before deciding to commit to it.

Authorities want the traditional customer-provider relationship where a trustworthy provider shoulders the responsibility the responsibility for debugging, support and future development, no matter if open source or proprietary software is used. It can be a valuable asset for an authority to be able to modify software, but it is avoided for off-the-shelf software in order to be able to guarantee stability.

The gained organizational transparency that open source code provides is of democratic interest, in particular of authorities. The Swedish Principle of Public Access allows all citizens of Sweden access to all records that are not classified. Why should this not apply to the source code used by Swedish authorities as well?

When reasoning in risk management terms, authorities should avoid being dependent on organizations abroad. It could prove to be disastrous in times of war, or other multi-national scenarios, when critical support cannot be provided. With open source, it is also possible to guarantee that no malicious functions for espionage or other functions violating the Swedish Personal Data Act.

Low or the absence of license fees is a common argument for open source software. But one has to recognize that the purchase cost only is a part of the total cost of ownership. There are large cost savings to be made. Currently are Swedish authorities purchasing solutions based on proprietary software from private companies. When the projects are finished, the companies are able to sell the same solutions again to other authorities for the same amount of money. Governmental funds could be much better spent if the authorities choose an open source solution and share the source code and experiences with each other.

Bibliography

[1] Khawar Zaman Ahmed and Cary Umrysh. *Developing Enterprise Java Applications with J2EE and UML*. Addison-Wesley Professional, 2001-10.

[2] AIIM. What is ECM. `http://www.aiim.org/about-ecm.asp`, Visited 2008-03-17.

[3] Alfresco. Enterprise Network - Open Source Enterprise Content Management. `http://www.alfresco.com/products/networks/enterprise/`, Visited 2007-03-17.

[4] Alfresco. Spaces - AlfrescoWiki. `http://wiki.alfresco.com/wiki/Spaces`, Visited 2008-03-12.

[5] Alfresco. Open Source Enterprise Web Content Management - Licensing Policy. `http://www.alfresco.com/legal/licensing/`, Visited 2008-03-17.

[6] Alfresco. About Alfresco - The Open Source Alternative for Enterprise Content Management. `http://alfresco.com/about/`, Visited 2008-04-28.

[7] Alfresco. Alfresco Solutions - ECM - Open Source Enterprise Content Management - Java Repository. `http://www.alfresco.com/products/solutions/ecm/repository/`, Visited 2008-05-08.

[8] Alfresco. Alfresco Announces First Embedded Enterprise Content Management System. `http://alfresco.com/about/`, Visited 2008-05-13.

[9] Alfresco. Roadmap - AlfrescoWiki. `http://wiki.alfresco.com/wiki/Roadmap/`, Visited 2008-05-15.

[10] Atea. Ementor ASA - Sweden - Atea Sverige AB. `http://www.atea.com/templates/Page.aspx?id=16597`, Visited 2008-04-28.

[11] Rebecca Barclay and Philip Murray. What is knowledge management? *Knowledge Praxis*, 2000-02-23.

[12] Toby Bell, Karen Shegda, Nikos Drakos, Rita Knox, and Debra Logan. Cool vendors in Content Management ID Number: G00146681. *Gartner Research*, 2007.

[13] TIOBE Software BV. Tiobe Software: Tiobe Index. `http://www.tiobe.com/index.php/content/paperinfo/tpci/`.

[14] Jackie Cadorin. Day Software Delivers Standardized Connectivity for IBM Lotus Domino. `http://day.com/site/en/index/company/press_center/press_releases/day_software_delivers4.html`, 2007-03-19.

[15] Mark Driver. Gartner Open Source Summit. `http://www.gartner.com/it/page.jsp?id=502444&tab=overview`, 2007-09-19.

[16] Carl-Fredrik Eld. Interview, 2008-03-18.

[17] Magnus Ericsson. Personal communication, spring 2008.

[18] Brian Fitzgerald. The transformation of open source software. *MIS Quarterly*, Vol. 30 No. 3, 2006 pp. 587-598.

[19] Peter Foberg. Interview, 2008-05-07.

[20] Swedish Armed Forces. Project Document and case management - Legal framework. Document was included in formal request HKV 2006-12-21, 09 621: 79944, 2008-03-13.

[21] Paul Holmes-Higgin, Luis Sala, and Roy Wetherall. Taming the Beast: CMS integration on the desktop with CIFS, Office, Dreamweaver and anything else. `http://wiki.alfresco.com/w/images/4/4a/OSCMS.pdf`, 2007-02-12.

[22] Tim Jennings. Defining the document and content management ecosystem. *Butler Group*, 2002-09.

[23] Marcus Jerräng. Staten storsatsar på öppen källkod - Computer Sweden. `http://computersweden.idg.se/2.2683/1.160841`, 2008-05-02.

[24] Gary Johnston and David Bowen. The benefits of electronic records management systems: A general review of published and some unpublished cases. *Records Management Journal*, Vol. 15 No. 3, 2005 pp. 131-140.

[25] Josh Lerner and Jean Tirole. *The open source movement: Key research questions.* European Economic Review, 2001-04-17.

[26] Linux Magazine. Top 20 companies to watch in 2008. `http://www.linux-mag.com/id/4766/3/`, 2008-01-10.

[27] Nelson. Open Source Initiative OSI - The BSD License:Licensing | Open Source Initiative. `http://www.opensource.org/licenses/bsd-license.php`, 2006-10-31.

[28] National Archives of Australia. Functional Specifications for Electronic Records Management Systems Software. `http://www.naa.gov.au/Images/ERMSspecificationsupdated_tcm2-1008.rtf`, 2006.

[29] Runa Patel and Bo Davidson. *Forskningsmetodens grunder: att planera, genomföra och rapportera en undersökning.* Studentlitteratur, Lund, 2003.

[30] Bruce Peren, Eric Raymond, and Ken Coar. The Open Source Definition (Annotated) | Open Source Initiative. `http://opensource.org/docs/osd/`, 2006-07-07.

[31] Jonas Persson. Interview, 2008-05-05.

[32] John Powell. Efficient, Cost Effective Software for Information Governance. `http://www.islington.gov.uk/Council/councilstructure/Access_to_Information/EdrmInThePublicSectorEvent/alfresco_software_for_info_gov_jp.asp`, 2008-02-22.

[33] Jenny Preece, Yvonne Rogers, and Helen Sharp. *Interaction Design: Beyond Human-Computer Interaction.* John Wiley & sons, Inc., 2007.

[34] Eric Raymond. *The Cathedral and the Bazaar.* O'Reilly, 2002-08-02.

[35] Andrew Savory. Interview, 2008-04-07 and 2008-04-08.

[36] Heather Smith and James McKeen. Developments in practice viii: Enterprise content management. *Communications of AIS*, Vol. 11 No. 33, 2003.

[37] Richard Stallman. *Free software Free society.* Free Software Foundation, 2002.

[38] David Vogel. Interview, 2008-03-11.

[39] Wikipedia. Proprietary software. `http://en.wikipedia.org/wiki/Proprietary_software`, Visited 2008-03-17.