RAPPORT 1211

Steinar Kristoffersen

SAFE AND ROBUST CONTENT DISTRIBUTION

Challenges and solutions related to Internet-based sharing of business critical documentation

MØREFORSKING
MOLDE

Høgskolen i Molde
Vitenskapelig høgskole i logistikk

Steinar Kristoffersen

Safe and robust content distribution

Challenges and solutions related to Internet-based sharing of business critical documentation

Report 1211

ISSN: 0806-0789
ISBN: 978-82-7830-174-6

Møreforsking Molde AS
August 2011

Title	Safe and robust content distribution. Challenges and solutions related to Internet-based sharing of business critical documentation
Author	Steinar Kristoffersen
Report number	1211
Procject number	2368
Project name:	Technical solutions for information sharing in the value chain
Project leader:	Steinar Kristoffersen
Source of funding:	Rolls-Royce Marine AS, VRI Møre og Romsdal
Report canbe ordered from:	Molde University College, library Post box 2110, 6402 MOLDE: Tel.: 71 21 41 61, Fax: 71 21 41 60, email: biblioteket@himolde.no – www.himolde.no
Number of pages:	50
Price:	NOK 100,-
ISSN	0806-0789
ISBN	978-82-7830-174-6

Abstract

We have studied challenges and some possible solutions for a demanding information sharing situation at Rolls-Royce Maritime AS (RRM). Rolls-Royce Maritime have a leading role in the North-West maritime cluster, with its equipment installed on 30 000 maritime vessels worldwide and international partnership with more than 2000 customers. Thus, it enters into many elaborate consortia and configurations of contractors working towards the delivery of a finished vessel, together with a varying and multifarious set of designers, yards, component suppliers, classification societies, auditors and consultants.

Rolls-Royce Maritime have to share documentation with all its partners, who are autonomous and numerous, and have independent and potentially incompatible information infrastructures. Some of these partners may already be part of, or may in the future enter into, other consortia with which Rolls-Royce are competing. Hence, information sharing has to be seen as a strategic undertaking.

This report discusses challenges of data protection, traceability, and version management as well as user administration overhead. It discusses one solution based on *rsync* in which the number of port numbers opened in the firewall is kept to a minimum. The solution relies on standard application software, open APIs and protocol, and hence it contributes to maintain a suitably loose coupling, in technical terms, of the partners involved. Moreover, it examines a replicated architecture from Syntergy, which would produce similar functionality in a packaged solution. Finally, an extension of the existing Open Text platform is suggested. Thus, we also contribute to the current research agenda in this area, looking at strategic issues as well as technological challenges of implementing a system of the type described.

PREFACE

Møreforsking Molde AS have been working closely with Rolls-Royce Marine AS in a project to investigate their options when it comes to document sharing over the Internet. This report documents the findings and experiences from that project.

The idea of the project started as an ambition to develop strategic and technical support for life-cycle management of large-scale, distributed and complex manufacturing of vessels for the off shore oil & gas services industry. It is symptomatic, we believe, for the problem domain that we thus addressed, that it was judged premature and overly complex for a small project with only two partners. Rolls-Royce Marine wanted this effort to be more solidly rooted in a wider consortia, and did not see it as suitable for a single company to establish a supply-chain wide project. Hence, we turned together, to the VRI foundation in order to solicit necessary support for a smaller, more concrete effort, which became this project.

In terms with the strategy and ambition of the VRI-foundation, we still see this project firmly as a step in the direction of supporting cross-supply chain collaboration in the maritime cluster with better understanding and technology in the future. We will use this report as actively as possible to recruit members into a bigger research consortium in the next instance.

This does not mean that this report does not stand firmly on its own, however. It tackles a problem that is rather more technical (and tactical) instead of strategic. Indeed, it may even come across as trivial. It is still a concrete and critical stumbling block for co-operative construction of some of our region's most central industrial products, namely the advanced maritime vessels for oil & gas as well as fisheries.

The industry of the North-West maritime cluster needs to co-operate widely to produce the ships that balance the requested quality and cost. This configuration of the value chain is, at the same time, quite flexible. New enterprises come and go. Suppliers deliver components to different yards, and the yards in their turn have the ships designed by various technical consulting companies. Thus, specifications, drawings and plans have to be shared across the value chain. At the same time, these artefacts represent the history and competencies of the company, which is their foremost competitive advantage. They need to be shared with care.

Shared documentation needs to be technically supported with forefront Information Technology (IT), since it is developed and managed internally, within each company, in this way. On the other hand, each company uses different technologies and formats, and even if they did not, the document repositories are protected behind firewalls, internal username/password combinations and cryptography. On top of the separating technology, there are no two companies doing everything exactly the same way. Therefore, elements and documents have non-compatible names, they are indexed and archives according to different principles and they are, last but not least, not used or valued internally in a compatible way. Proposing the bridge to cross this gap is the scope of this project and the report.

Steinar Kristoffersen, *project manager* Molde, August 2012

TABLE OF CONTENTS

1 INTRODUCTION

The ship building industry is documentation-intensive. Many parts of the manufacturing process involve components, equipment and technologies that have to be specified and managed in an orderly way. Tracing, tracking and distributing documentation becomes an integral part of the engineering and construction work, which again is globally distributed. This means, for an efficient organization of the work to be implemented, that a document management system needs to be in place.

Rolls-Royce Maritime (hereafter called RRM in this report) uses the Open Text platform for its document management applications. It has been supplied and maintained by *Contesto AS* (http://www.contesto.no), alongside with internal resources of the RRM's IT-department.

This is not an infrastructure that is used by all RRM's partners, nor is RRM in a position to be able to enforce supply-chain wide adoption. This means that standardization initially seems to have limited potential, and alternative strategies are needed in order to be able to share information outside the preferred information infrastructure.

The challenges that we were given by RRM's IT department were to suggest and describe an implementation strategy to support the exchanges between the internal document repository thus implemented with Open Text-technology, and various other partners' IT-infrastructure . There were few constraints given, except requiring minimal technology intervention at the customer's side, simply because the coupling of collaborative partnerships in this kind of industrial cluster is quite loose.

The maritime business of our region (and beyond, clearly) needs to work together to produce advanced ships for different sectors, most noticeably the oil and gas services and fisheries. The actors of this supply chain balance requested quality, flexibility and cost through specialization and a flexible association of suppliers. The suppliers deliver components to different yards, and the yards in their turn have the ships designed by various technical consulting companies. Thus, specifications, drawings and plans have to be shared across the value chain. At the same time, these artefacts represent the history and competencies of the company, which is their foremost competitive advantage. They need to be shared with care.

This does not necessarily mean that there is no trust within the supply chain, usually; it is probably more often exactly the opposite. Many actors might probably have quite preferred not to have to track and maintain a DRM (digital rights management) scheme for the information they have flowing outside of their own enterprise network, if only to reduce costs. There is a formal as well as social aspect of trust, however, and the former entails for partners to be able to implement and account for an orderly process, if that were to be required. Quality assurance audits are common, and across the supply chain technical artefacts, such as the construction drawings and parts lists, are modified and extended. This needs to be reflected in official versions of the documentation, to avoid deprecated material to be circulated or re-used later. This means that version management and sound "return re-engineering practices", which in other engineering disciplines is sometimes called *incremental consistency [1]* , needs to be encouraged.

The *sharing* needs to be technically supported with forefront Information Technology (IT), since it is developed and managed internally, within each company, in this way. Email or memory stick-based exchanges will not sustain the tracing and tracking of multi-versioned documents sufficiently well. On the other hand, since all the collaborating partners use different technologies and formats (and even if they were not, the document repositories would be protected behind Firewalls, internal username/password combinations and cryptography), the common ground may be difficult to establish.

On top of the explicitly separating technology, such as firewalls, incompatible platforms, there will be no two companies doing everything exactly the same way. Therefore, elements and documents have non-compatible names, they are indexed and archived according to different principles and they are, last but not least, not used or valued internally in a compatible way. Proposing a conceptual and technical “bridge” to cross this gap is the scope of this project and the report.

1.1 Research agenda

The goal of this project is to identify alternatives and recommend a robust and secure solution to document replication from Rolls-Royce Marine’s internal network repositories in an Open Text Content Server (CS) solution to collaborating partners in its global project setting.

There are several approaches to shared access or replicating repositories using the CS itself, but most require direct access to the infrastructure and equivalent licenses and software solution on both sides of the “bridge”. Initially, the mandate stipulated that we try to avoid this. Also, in a long-term perspective, it is better not to become dependent and “locked into” specific vendor technologies, as well as formats and document types.

The proposed solution must work equally well with unstructured content, communication and documents, as well as database records.

1.2 Project requirements

The following sections summarize the requirements to the project.

1.2.1 Safe and robust sharing of documents from inside the firewall architecture

The solution needs to be able to take documentation in various formats, from inside the CS of RRM, and share it with customers and partners over the Internet. We may be able to satisfy this requirement using “hub architectures”, which entails servers controlled by either party or shared exclusively by RRM in some kind of DMZ (”demilitarized zone”), or in a totally distributed fashion.

1.2.2 Controlling distribution and versioning of documents across systems

It is a desired feature of a document synchronization that is comprises some ability to control and check the distribution and life-cycles of documentation, in other words that some tracking and version control is implemented. There are of course many possible levels of such control, which could comprise any combination of:

- A complete version managing system, in which documents are checked out to designated and named partners and thus placed under a “quarantine” that means it may not be changed elsewhere until it is “released”.
- A less restrictive sub-versioning scheme, in which modification to documents in parallel is allowed, but lead to additional branches and releases of a given document.
- Assigning a “time-to-live”. This would mean that a document might have local copies which are treated as the “original” until a set time/date, after which it is no longer judged part of the distribution. Any subsequent changes to it must then be incorporated manually.
- Implementing a degree of DRM, in other words a system by which illegal copies of a document (or copies obtained legally, but retained after their “time-to-live”), either cannot be open or will be watermarked in such a fashion that they may be traced and identified as RRM’s property.

1.2.3 Gain experience and recommend a solution

We have also been using this project, together with RRM, to gain practical experience with existing technologies and approaches to document sharing. Some testing was to be carried out, albeit in a limited scale, so that the recommended solution could be presented as viable for the existing organization.

2 RESEARCH BACKGROUND

This chapter briefly summarizes the data sources and discussion partners that we have had, outside of the many meetings with RRM which took place during the project.

It was, from the beginning of the project, clearly expected from us that we tried to use or at least seamlessly integrate with Open Text software as much as possible. Therefore we also were to establish contact with the company that have been RRM's partner for this package.

We have had several telephone conversations with sales engineers at Contesto (http://www.contesto.no/about-contesto/), which is a small supplier of information management solutions.

Our discussion resulted in no further or different conceptual design, compared to our original ideas, rather, they contributed to confirm one of the hypothesis which emerged from out research, namely that basic operating system utilities (such as *rsync)* is also the foundation of allegedly more sophisticated software offered to the enterprise market. Hence, some kind of rsync-based solution will probably work well and seem to meet the needs of the company. It illustrates a certain trade-off between flexibility and functionality, however, which we will return to.

There seems to be no established standard solution or architecture for inter-supply chain documentation sharing. Asking a few questions to other companies in the same segment reveals that many companies rely (or rel*ied*, at the time, since this is an area of continuous change) on technological practices that would not have met the requirements that we aim to meet for RRM.

For instance, we have been informed that one company, well known in this region, uses email a lot to send documentation, whilst another big, international company relies on a combined strategy comprising their own proprietary document management system, which again may be tailored to individual projects. From what we were told, they try to get direct access to partner site servers, and aim to control the entire document logistics chain. This is probably a quality assurance measure. They also provide external access to this system if necessary. Both these example companies filter design documents so that models cannot be engineered easily from them.

3 DETAILED REQUIREMENTS' SPECIFICATION

3.1 Protect sensitive information

Rolls-Royce Maritime projects are going to involve many partner companies and individual consultants that simultaneously or shortly after may be partners of other projects involving Rolls-Royce's *competitors*. This entails a cautious strategy with regards to publishing sensitive material, both with regards to the number and background of recipients and the possibility to track subsequent distribution.

3.2 Keep additional overhead low

Even if partners and customers were to be allowed access to the RRM internal network or CS via a VPN (Virtual Private Network) the administration of usernames and passwords would represent a substantial overhead. Hence, initially, direct access to the internal CS was not an option presented to us. The requirement is for RRM internal overhead for user administration and maintenance is kept at its current level. We have interpreted this as meaning that the additional components shall be manageable by the current staff and that external user registration is not acceptable.

For partners and customers, some work may be acceptable to be able to retain information and partake in the document flow from RRM, since they already have to take technical steps in order to receive it. However, it should be comparable to what they already need to do; it would be preferable if it did not require additional software installation and asset management on their side.

3.3 Workflow and version management need to be maintained

All RRM projects comprise complex technical engineering processes, within which specifications, designs, drawings and reports are developed over time by many contributors. Some degree of version control and workflow support therefore needs to be maintained. This goes beyond, and is partly independent of the technical approach to keeping variants and version apart, and involves the working practices which the documents are used to guide, govern or represent as processes and products. From the perspective of RRM, this work may be supported by the workflow module of the Open Text CS, in such a fashion that certain steps are required to be taken before or after achieving a document's "next state". It is a requirement that the recommended solution may be part of the RRM workflow system.

3.4 Allow loose coupling in supply chain

Since the various partners who are involved in a project are distinctly different by nature of their contribution, history, ownership and technology, no assumptions may be made about standard policies, processes or formats. There is, on the other hand, some element of *de facto* standardization between the companies, which is necessary for them to work together in the way that they already do. Thus, some 3D-drawing systems (CAD), Microsoft Word and

Adobe Acrobat (PDF) are already common across the companies. However, they categorization/indexing terms vary, and so do the encompassing work practices.

3.5 Keep cost of implementation, licensing and maintenance low

The proposed solution should of course be sufficiently cost-efficient, but also take into consideration the role of RRM within the Rolls-Royce group. The replication platform that the project ends up recommending will be looked at as either a separate initiative or a pilot proposition coming out of RRM, and therefore it needs to scale well and be aligned with RR overall technical strategies.

The basic challenge, therefore, is to implement a shared repository outside of the DMZ (De-Militarized Zone), in such a way to it is easily managed and does not compromise the document control that is already implemented on top of an Open Text Document Management System. It should be technically, organizationally and economically lightweight, yet efficient. This means also that for partners and customers, little extra effort and no enforced change to their own information infrastructure should be necessary.

3.6 Additional requirements

There are some additional requirements that have been discussed in the initiation phase of the project; they are also important, but may be seen as instrumental supplements, in a technical sense, to complying with the main requirements asked from the project.

- The solution should support encrypted transmissions.
- Data-integrity checking against tampering would be an advantage.
- It ought to maintain current firewall regimes, opening as few port numbers as possible.
- The solution should to maintain a robust and predictable treatment of erroneous or executable content.
- It should be security-aware, in the sense that RRM employees, customers or partners may not easily be tricked to try to login outside of the DMZ, at a fraudulent site.

4 TECHNOLOGY SURVEY

There are a large number of technology-centric solution candidates, many of which are supplied and supported by commercial companies, which could have been chosen as a mediator of content from the existing CS. They are listed below, and may be surveyed in more detail at a later stage of the process that is described in this report. It was, however, explicitly asked from RRM that we delayed this activity until the end of the project, if the resources were not already allocated elsewhere, e.g., to implementing an in-depth testing a one or a few alternatives:

- SharePoint
- OpentText Integration Center/Business Integration Studio
- Vignette Integration Studio
- Sitrof Doc Exchange
- eCo
- RosettaNet
- BizTalk
- cXML
- MESChain concepts
- Completion.no, developed and managed by Novomar AS
- MyProject, developed and managed by Wise Consulting AS
- Dropbox (http://www.dropbox.com/)
- Syncplicity
- SyncToy
- SyncBack
- Syntergy
- Repliweb managed file transfer
- PeerCollaboration/PeerLink
- GoodSync

Next, the commonly available middleware and open source platform technologies, thus labelled to indicate the need for more involvement from RRM's own IT-resources in order to be set up correctly, monitored and supported in the future, are also and correspondingly listed for reference.

- Plain SVN (Subversion) or a similar CVS (Concurrent Versioning System)
- Rsync on top of ssh and, for which rsync is used interchangeably in this report, rsync-ports and extending systems, such as
    - Unison *(*http://www.cis.upenn.edu/~bcpierce/unison/index.html*),*
    - Lipsync, which is a wrapper around the most common unix protocols and utilities, such as *ssh* and *rsync* (Figure 1), and
    - DeltaCopy, which is a similar type of solution, with Cygwin's protocol stack
- Syncrify, which has its own native stack implementation of *ssh* and *rsync.*

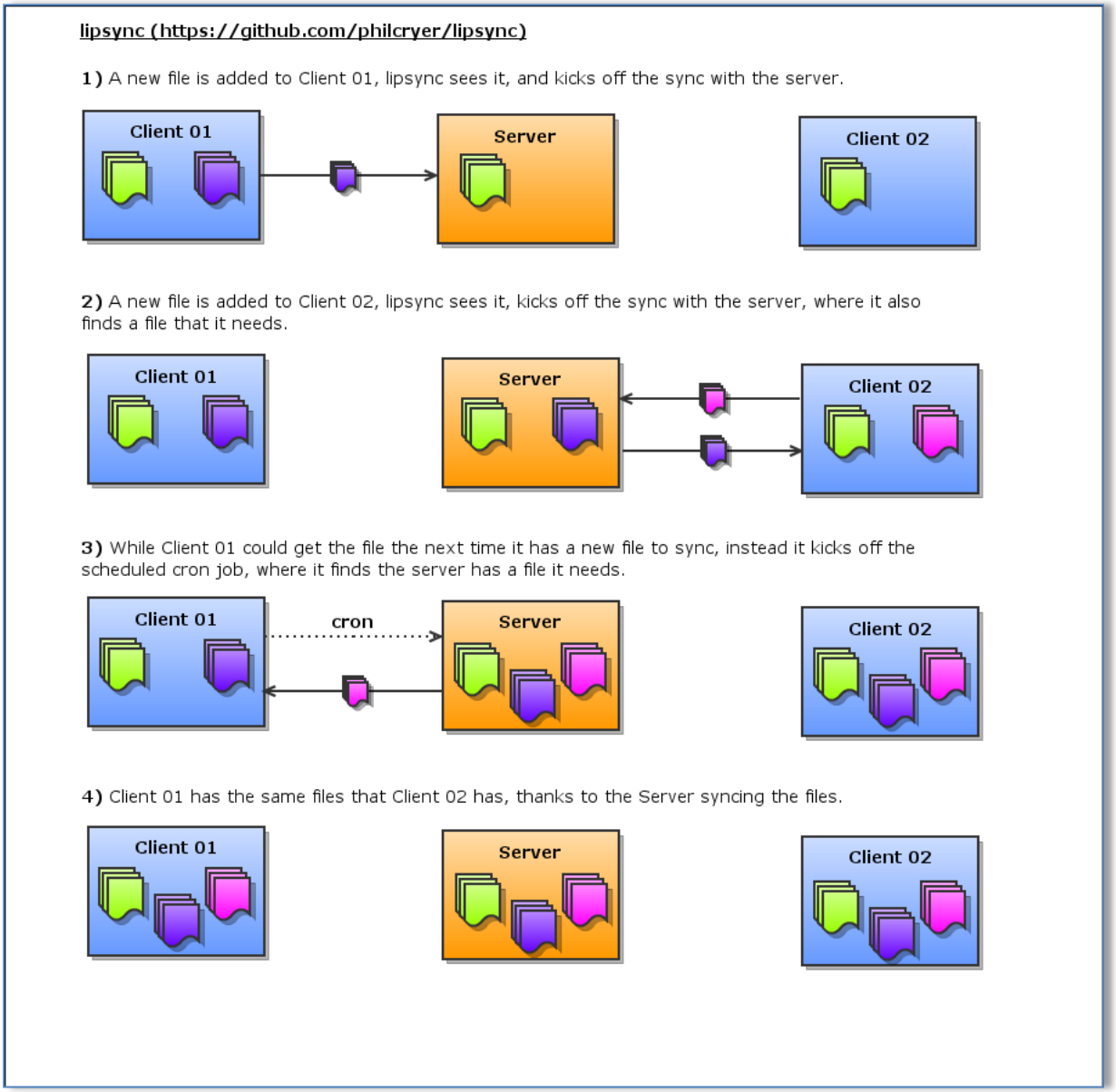

Figure 1: Lipsync conceptual model of interaction between hosts

Finally, one main candidate from each category is short-listed initially, by merit of its simplicity and robust fulfilment of most of the requirements that were discussed above.

- *Rsync* on top of *ssh*, onto an authoritative web host outside the DMZ
- Syntergy Replicator, which provides a full multi-directional replication capability. CS objects are replicated to one or more participating systems, and modifications are reflected back to every other copy as well as the original object.
- Open Text Document Management System Email-based re-distribution (eventually with email return using eLink).

We have also been running several tests, for instance using DeltaCopy and Syncrify.

Rsync has the advantage of being highly flexible, since it is configurable and scriptable in accordance with what one would expect from a Unix utility. In this report, we treat it as a representative also for many of the tools that are built on top of the pertaining protocols, such as Lipsync (Figure 1). Syncrify and DeltaCopy are other tools that are quite similar to Lipsync, inasmuch as they offer a graphical user interface to configuring and monitoring the services. There is a minor, technical difference between the tools, which we will only in passing refer to in the discussion later, and that is that:

- LipSync (https://github.com/philcryer/lipsync) runs on Linux 2.6.x (Figure 1).
- DeltaCopy (n/a) relies on the Cygwin implementation of rsync and ssh for Windows (Figure 2)
- Syncrify (http://www.syncrify.com) implements its own corresponding protocols, so it can run on MS Windows, Linux, Mac OSX and UNIX. (Figure 3).

Next, the preliminary candidates are looked into in more detail in order, firstly, to elicit the necessary areas warranting further investigation.

Figure 2: DeltaCopy cygwin-based stack

Figure 3: Syncrify native stack

5 THE OPEN TEXT DOCUMENT MANAGEMENT SYSTEM WITH EMAIL RE-DISTRIBUTION

The Open Text Document Management system, like most commercially sensitive system, admittedly, is only superficially documented on the company's website[1], and hence it is difficult to describe it in a detailed technical manner.

We have conceptualized the system according to the whitepapers that the company has published about it, as being a centralized repository for documents of all kinds. It has its own internal database for the documents, by which it may index content with metadata to make it more easily retrievable.

5.1 Email publishing

From the documentation[2], it seems quite straightforward to configure Open Text (presumable also in the version already installed at RRM), to be able to provide the bridge that we are looking for in this project, at least so that the documents which have been checked out may be received back into the system, by:

- Sending documents from within the repository as attachments to emails targeting arbitrary receivers. Email copies of files from the repository may this be made available upon explicit request, workflow state changes or as a scheduled exchange.
- Assigning unique email-addresses to folders in the document management system, by which the Open Text platform may receive documents and store them accordingly, with consistent version management and bookkeeping of authoritative copies as requested.
- *The options available to designate email message headings and stipulate the return path needs to be looked into more carefully for this option, since it requires:*
    - *Some intervention either programmatically or manually will be needed to select and rename documents going out from the repository.*
    - *A descriptive naming scheme needs to be organized, so that documents are marked-up with information designating their return-path.*
    - *Based on the documentation and discussions with Contesto, we have learned that folders rather than files may be so-called eLink receivers.*

5.2 The eLink technology of Open Text

eLink may be set up to work with projects, folders, compound documents, and any type of container that may be customized from these types. Once a container is configured to be managed by eLink, all documents may be emailed into it without logging into the Content Server as a registered user. When a document is added to the Content Server via email, the

[1] https://www.opentext.com
[2] http://www.opentext.com/2/global/products/products-opentext-document-management.htm

Content Server itself determines whether the document already exists or if it is a new instance of that document. If the document already exists, the server commits a new version of that document. If the document is new, it is added. When enabling a container for eLink, one has to decide if just the attachments are going to be added, or the entire email messages. Unlike the *Discussions* feature, eLink has no option to subscribe to content[3]. From the sources we have had available, it has not been possible to disclose the exact functionality of the Discussions functionality in Open Text, but it may seem that it provides an opportunity to publish (or at least notify) subscribers about changes in documents through subscriptions.

There may be some challenges concerning the distribution of documents via email, of which we shall deal with two in this document:

- Legitimate revision and return of the received document, however, not using the stipulated return-to-address, but instead to the internal project contact at RRM or a potentially arbitrary sender of the original message onto which the document was attached. In these situations, the document would have to be re-routed back to where we wanted to receive it. This reservation may also be dealt with by configuration of the Open Text email-filing tool[4], which seems to be able to automatically check the content of the Microsoft Outlook folders and compare it with the content in the CS library. Hence, even stray document copies that have found their way into the ordinary inboxes of RRM users involved in the project, may be caught.
- Rogue re-distribution of documents via local storage at the remote side, eventually after modification or re-engineering of the document. This is not an easily solved problem, regardless of the chosen approach. However, enforcing circulation through the receivers' email system ensures a relatively efficient auditing process at the remote site. Reception of the email and initial storing of attachments is likely to leave electronic tracks, the awareness of which may serve as a mild reminder to be conscious of further distribution.

It is an important concern associated with this approach, however, that illegitimate document copies or types may be returned into the document repository by anybody who has managed to get access to a folder's unique email address. This may, however, be dealt with by programmatically at a sufficiently frequent rate, and making sure that the time window in which a certain email address is valid remains brief and pertinent.

A supplemental service index may be hosted remotely. Thus each document designated to be re-distributable via email/eLink, may keep an association between the file name and a *mailto*-tag. This way, the external user will appear to have a local copy of the document, whilst RRM could implement pass- and pin code-based authentication schemes to get the *current* name of the file (leaving other names void), and hence also the opportunity to customize mail headers and other mark-up data for the return email as well as the outbound

[3] Open Text eLink QuickStart Guide
[4] http://www.opentext.com/2/global/products/products-law-firms/products-opentext-edocs-email-filing-law.htm

messages. For the volume of messages and affiliated documents that we are concerned with, however, it may be a viable option, indeed, to deal with return copies manually or via the automatic filing system described briefly above.

Main advantage: The *email*-infrastructure is ubiquitous, platform-independent, robust and simple. It requires in a minimal implementation, little intervention and no new technology.

Main disadvantage: Return-to email addresses may become widely known and used maliciously, for bad content or to launch primitive "Denial of Service"-attacks. Email attachments may easily become too big.

5.3 Workflow distribution

The documentation from OpenText also confirms that it furnishes functionality that may incorporate workflow and document management in an event-driven fashion, and hence, email may be used to distribute files programmatically from the repository as well. This needs to be fully verified, however.

5.4 Genio

We alternative, which relies on technology that has been largely unexplored due to lack of support from the supplier representative in Norway (Contesto), but we think that it is important to mention it regardless. It comprises using:

- A native integration engine, Genio, which may be installed on the DB (Document Broker) as well as at the customer side. The interaction design might be conceived as follows:
  - If the Genio (or similar infrastructure-wide integration tools, like BizTalk, etc.) has been installed with a client at the user side, the documents may be accessed directly at the local file system.
  - Otherwise, if a "communcal" license and sandbozing-regime is to be implemented, the user logs in like in alternatives 1 and 2, to find the links on the DB of the files which have been scheduled for Genio-replication onto the broker.
  - Upon upload, the files are replicated back by Genio, either from the user site directly or the "sandboxing" DB.
  - Users' identity may be transparent (passed-on) or local to the DB (with only one super user to the CS, in which case a separate user record is needed on the DB, like in alternatives 1 and 2).
- If the integration engine is to be installed at the user sites:
  - User identity management must be transparent end-to-end, so Rolls-Royce is required to administer all the users

- Genio will require more port numbers to be opened.
- Genio executes SQL on the host inside the internal network.

6 *RSYNC* ON TOP OF *SSH*, ONTO AN AUTHORITATIVE WEB HOST OUTSIDE THE DMZ

Rsync represents in many ways a technologically minimalist point of departure, which on the other hand (or consequently) draws upon a little more extensive technical competency in order to be set up correctly and work well. On the other hand, this type of solution is highly transparent, in the sense that the local IT-department at RRM will be in total control of the installation and have complete insight into how it works.

In the transparency aspect, as well as many others, the rsync-based approach is the opposite of the formerly discussed Open Text email-based protocol. For instance, it will truly replicate files from within the DMZ/CS onto an external server, of which hosting and management may be completely outsourced or even installed and configured to serve dedicated partners in a project. In fact, physical servers may be shipped and returned on project demand, or an operating system image distributed for the purpose across the Internet. Thus partners have easy access to a synchronized repository, which is adapted to their needs exactly, without having to engage in the implementation of the system. Rsync runs well on the least expensive hardware and with the freely available Linux operating system.

6.1 Rsync basis

Rsych is Secure Shell (SSH)-solution, which is a protocol that we know to be well integrated with Open Text distribution and integration systems already:

"Secure Shell is a transport protocol that allows users to log on to other computers over a network, execute commands on remote machines, and securely move files from one machine to another. It provides powerful authentication and secure communications over insecure channels, and is intended as a replacement for rlogin, rsh, and rcp. By using Open Text Secure Shell, administrators can eliminate the act of eavesdropping on sensitive information such as user credentials [5]."

A definite, necessary prerequisite in order to able to use this approach is that we may gain access to Open Text content on the files and folders level.

Since a direct, transparent access is not possible, an intermediate step may have to be scripted (**Feil! Fant ikke referansekilden.**), which increases the work needed to set up the solution, however, it also provides an opportunity to sandbox a replication bridgehead away from the CS proper, which will make the solution potentially more secure.

6.2 Rsync configuration and operation

Rsync is a versatile backup/mirroring tool, offering many configurable features. It is often used to synchronize content between development/test-environment and production

[5] http://connectivity.opentext.com/common/files/Data_Sheet_Open_Text_Secure_Shell_14.pdf

servers, which is comparable to the requirements' setting of this project, and to backup key areas of file systems both automatically through *cron* and by a CGI script.

These are some of the key features of rsync:

- It offers support for copying links, devices, owners, groups and permissions
- Rsync comprises *exclude* and *exclude-from* options (similar to GNU tar).
- It has many options for dealing with temporary or binary files
- Rsync does not require root privileges, but relies on the SSH-protocol to access the network resources. It uses privilege separation, which means that the SSH daemon running with root privileges accepts the connection to port 22, but delegates further network traffic to a non-privileged sub process.
- It uses pipelining of file transfers to minimize latency costs, which means that excessive transfer-*from* capacity is buffered for a parallel transfer-*to* process, to avoid idle time on the faster host.
- Rsync also has support for anonymous, user/password-logins (with or without domain users), registering authenticated rsync servers, which is ideal for mirroring and include/exclude-host strategies
- This technology offers passwordless authentication, which is possible using ssh-keygen. It can create RSA keys for use by SSH protocol version 1 and DSA, ECDSA or RSA keys for use by SSH protocol version 2. This type of cryptographic public/private-key security is normally sufficient for the public Internet.
- Rsync, rather than the whole files, transfers only actual changed pieces of files. This makes updates faster, especially over slower links like satellite modems. FTP would, in comparison, transfer the entire file, even if only one byte changed.
- All transfers are compressed and encrypted.

6.3 Using Rsync[6]

6.3.1 Setup

The receiving machines (mirror) must be configured to be a "rsync server" by running rsync in a daemon mode ("*rsync—daemon*" at the commandline), which is governed by a configuration file (*/etc/rsyncd.conf*).

Any number of machines with rsync installed may then synchronize to and/or from the machine running the rsync daemon. Moreover, one machine can synchronize towards many different mirrors, with different selection rules covered by the local rsync script.

Since RRM servers (which are the local hosts in this description), will be *behind the firewall and inside the DMZ, synchronization cannot straightforwardly be initiated from the outside*, e.g., using triggers from the *inet* damon. There are other options, however:

- Synchronization may be initiated by a local agent with email-receiving capacities on the RRM network, or similarly:

[6] http://everythinglinux.org/rsync/

- Remote collaborators may request an update via a web-site or directly from their mobile phone by sending an SMS to a gateway hosted by RRM, which in the next turn initiates an *rsync* update
- A *crontab* entry or other type of job scheduler on the local network may request rsync updates to be run at regular intervals.

In the Rsync config-file, a list of “inner” user password combinations may be set, which will limit the access to the server via the rsync daemon further. This does not have to de an LDAP or domain-level registered user name.

The “hosts allow” and “hosts deny” options for the (temp) path need to be exploited fully, to avoid undesired access, bearing in mind however, that we are talking about access to the mirror and not the main, internal server on the firewalled RRM network.

Rsync may then be run at the server with a script or command-line command, such as e.g.:

```
rsync—verbose  --progress—stats—compress—
rsh=/usr/local/bin/ssh  --recursive—times—perms—links—
delete—exclude “*bak”—exclude “*~” /www/*
webserver:simple_path_name
```

This example synchronizes the delta from the previous execution of Rsync on the local folder /www/ onto the webserver host (which may be nicknamed in /etc/hosts), including all permissions and timestamps.

6.3.2 Requirements fulfilment using *rsync*

Rolls-Royce Marine has stipulated that single/centralized repository architectures are sought, in such a way that conceptually the shared documents are:

- Kept in a robustly and provably official version in the internal CS.
- Mirrored or indexed remotely, possibly by a single host.
- Single-instance, which means that there may be many copies, but only one original, (as it were).
- It might not be necessary to look at advanced concurrent version management systems, since the documents will not be changed at several sites independently.

This final point is still important, though, and a significant difference between rsync and alternatives that exist “off-the-shelf” in the market. Rsync does not have built in conflict resolution and version management. A newer file will either overwrite an older one, or keep it. It is crucial to ensure that collaborators may be able to verify that they are looking at the latest, correct version of a document, or at least detect that there may be inconsistencies. Alternatively, one might integrate rsync with existing concurrent version management or configuration control systems, which on the other hand, would make it even more demanding to install, configure and maintain.

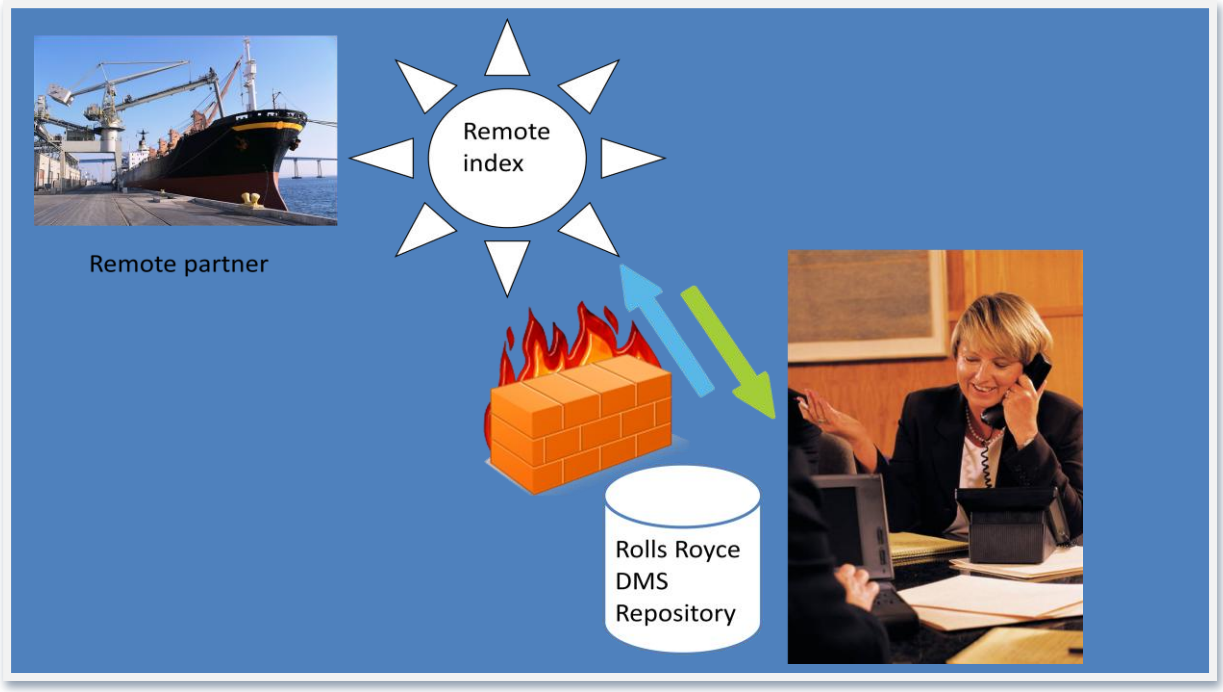

Figure 4: Single repository, centralized replication

Moreover, the design of a solution based on a mirrored central host means that the Rolls-Royce repository is kept entirely intact, and the bridgehead locally or all the way into the DMZ may be seen as a natural extension of the functionality that is already furnished by Open Text. Significantly, it allows Open Text (via Contesto AS) to be the main developing partner for the intranet part of the architecture, if that were to be judged desirable.

At the same time, it allows multiple, parallel solutions to be implemented on top of the mirror, on the interface towards collaboration partners. This means that there is room for each of the partners to contribute or request services that they need to perform well on their side of the supply chain.

A single, mirrored repository has many advantages

- It is easier to keep updated
- Less expensive and risky in terms of supporting it
- Data mining with one tool and one license is more manageable
- Access and role control is simpler

Thus, a solution based on simple file replication meets most of the requirements to a large extent. The solution based on rsync, on the other hand, will need to extract content from the Open Text CS using e.g., the W*orkflow* functionality. It may complicate matters technically, plus, it leaves an unmanaged copy of the file “checked out” from the CS. We presume that the workflow will be able to flag the content as having been exported, until it is imoprtet back in and hence may be “unflagged”.

In technical terms, however, replicating the extracted content across the using ssh-based *rsync,* will have the following technical footprint, which are aligned with the requirements:

1. It may run with limited authority, thus we may run daemon as non-root.
2. Only one port is kept open *in* through firewall, defaults to 873 for *rsyncd*
3. Port 22 used for ssh-transport *out, plus,*
4. Http-port (80) is used if rsync is invoked via an *inetd* daemon. This is not the only way, however, it is possible to schedule it using a *cron*-type of mechanism.
5. Port tunneling through DMZ or using the firewall is possible.
6. Cryptography and public/private passkey-pairs are available.

Rsync-based solutions may thus be integrated with the Open Text CS, and the return channel may be quite independent from its publishing, for instance by emailing it back directly to dedicated folders, using secret names with limited “time-to-live”.

Rsync works on the calculated difference between files, eventually with additional timestamping. This may be problematic, though, and reason to heed a warning: The file's data will be in an inconsistent state during the transfer and will be left that way if the transfer is interrupted or if an update fails. This is particularly problematic for binary files, of course, which may then not be possible to open again.

One of the characterizing aspects of the solution is that it is *de facto* already adopted by many of the existing systems that we have looked, which may be found as *open source*, platform-independent utilities, commercially available off-the-shelf software, and, presumably, if and when ordering tailored integration project from a consultant.

The advantage of building an in-house solution is that RRM gains flexibility and transparency, whilst on the other hand buying it off-the-shelf yields robustness and simplicity.

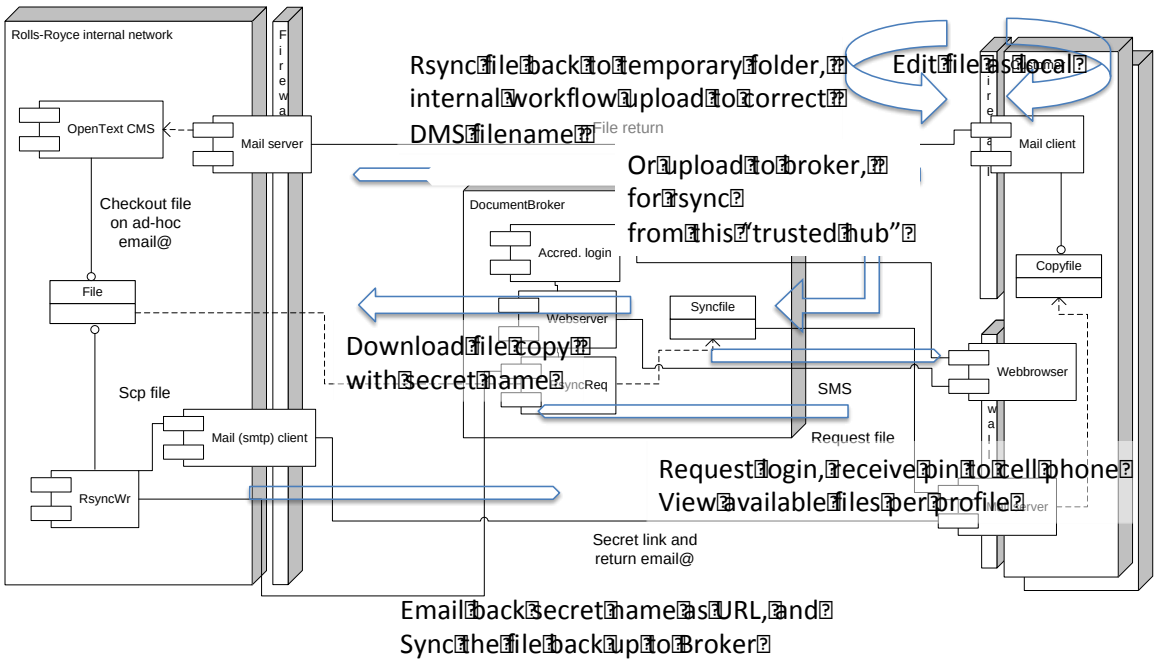

Figure 5: Alternative 1: Two-way rsync-based replication

We may summarize the interaction design of a rsync-based solution as follows, with some additional functionality, e.g., to generate a temporary passcode, which is published to the users' cellular phones:

1) The external user request login privileges from a web site, which may be hosted by RRM or the Document Broker (DB) service.
2) A SMS is returned with a temporary access code.
3) Entering the access code, the user is allowed to request a list of available file names, eventually alongside with their path/folder association.
4) The file name list is displayed in the browser, allowing the user to select and request files. The displayed name need not be identical with the internal id, which may be kept secret for security.
5) A crucial step, next, is to export the file from the CS, rename it either simply to provide a limited "time-to-live", i.e. a designated scope for the requesting user, or an eLink-folder that later may receive the file. A script executes workflow and *rsync* to publish the file under its secret name, usually using only port 22 (ssh).
6) Next it emails the secret name to the external user who requested it. In both contexts, the workflow functionality of the CS must script export/import, eventually via authorization by the project leader or document controller.
7) Replicate the file to the Document Brokerage service, rename if not temporary already.
8) Email the link to the file, using its "secret name".
9) The user may now (and until the RRM workflow, eventually, uses rsync to delete it from the broker, which is an action which of course also may be scripted, *crontab-ed* or governed using the workflow system) download the file and:
10) Edit the file locally. In case of re-distribution, it would be advisable to implement some sort of watermarking of the outbound files.
11) After finishing the work with the files, the external user may either rsync it directly back to the RRM CS, if that is available using a VPN (virtual private network) or via DB if that is allowed to enter through port 873 of the internal network (Figure 7), where it is picked up by workflow and returned to CS.

One alternative variant, which makes the return of modified files simpler in the sense that the workflow agent program is not required to verify and upload the system to the correct version relative to the CS is illustrated in design alternative 2. It uses the Open Text eLink functionality to automatically receive and upload files that are received by the designated mail server. In this design, each checked-out file is put into an eLink-enabled folder, which is linked to a unique email address. The mail server returns the files that it receives on this address to the originating folder, from which it is uploaded to the CS. The procedure followed by the system, at least on a conceptual level, is described in the interaction design diagram in Figure 6.

The biggest drawback of alternative 2 is that many customers, partners and RRM alike are likely to have upper constraints to the size and type of email attachments that they will send or receive. There is also, albeit much more unlikely, a risk that alien traffic may be addressed to the file folder, which may saturate or confuse the CS.

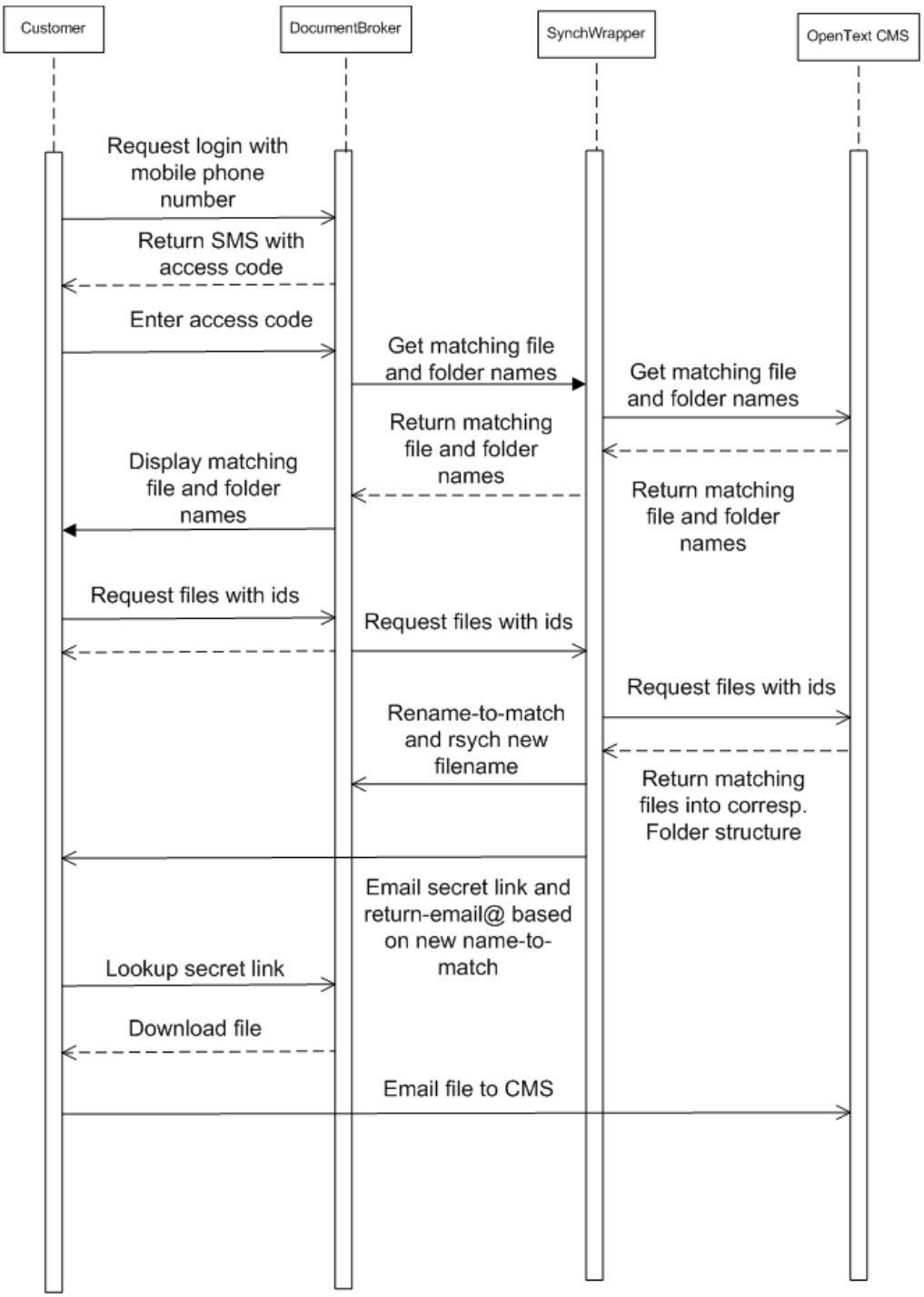

Figure 6: Interaction design diagram for design alternative 2

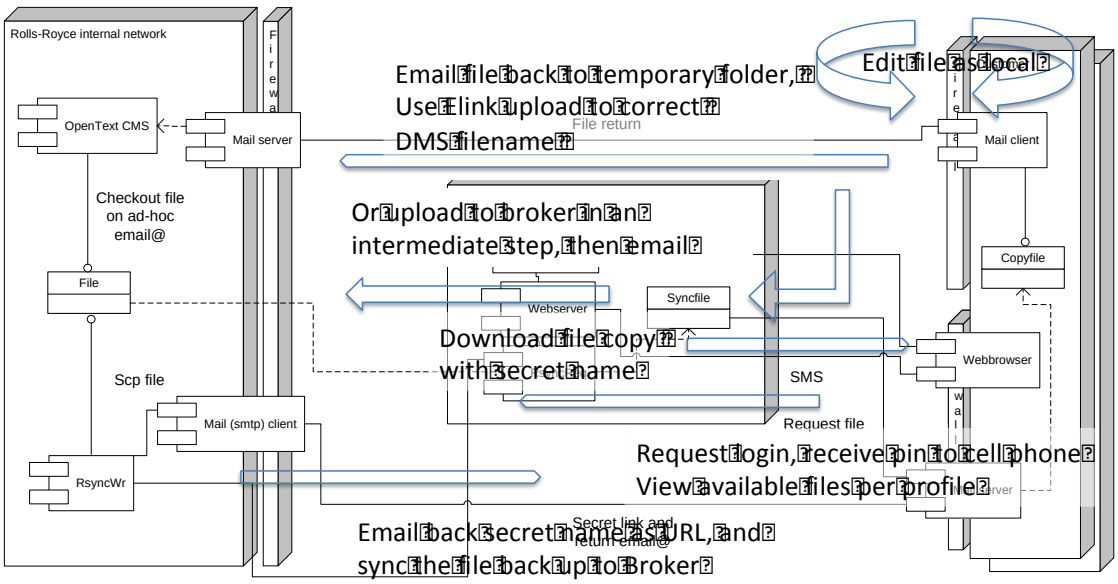

Figure 7: Alternative 2: Rsync publishing, return with email eventually via broker

- Alternative 2, in other words, starts like 1, but the workflow puts file in eLink-enabled folder.
- There is an opportunity that the files may be eLinked *out* as well
- The file/folder structure of the CS need to be tailored to the scoping needs of the system, otherwise the receiving folder name may become too widely known. This is a trade off. On the one end, designated “secret” folders may be ad-hoc generated to hold the files. Thus, only one user/timeslot-combination will have access to the folder and hence be able to return the file to the system. It involves more programming work, but better security. On the other end, a general project folds holds all exports, in which case version management will be internally robust, and less programming work needed, but many users may be able to return the same (or different) files so that external inconsistencies arise and must be resolved manually.
- After the user has edited the file,
  - It may be emailed directly to its secret filename@rolls-royce.com by the user
  - It may be uploaded to the document broker, which emails it back to the eLink-enabled folder, in order to deal with attachment size/type-constraints. For instance, the broker may be able to re-assemble files that have been split into manageable chunks by an external system.
  - The document broker may also remove or check watermarking, etc.
- Port 873 does not need to be opened, like in the first alternative, but large files may have to be dis- and re-assembled by software that is deployed on the users’ side.

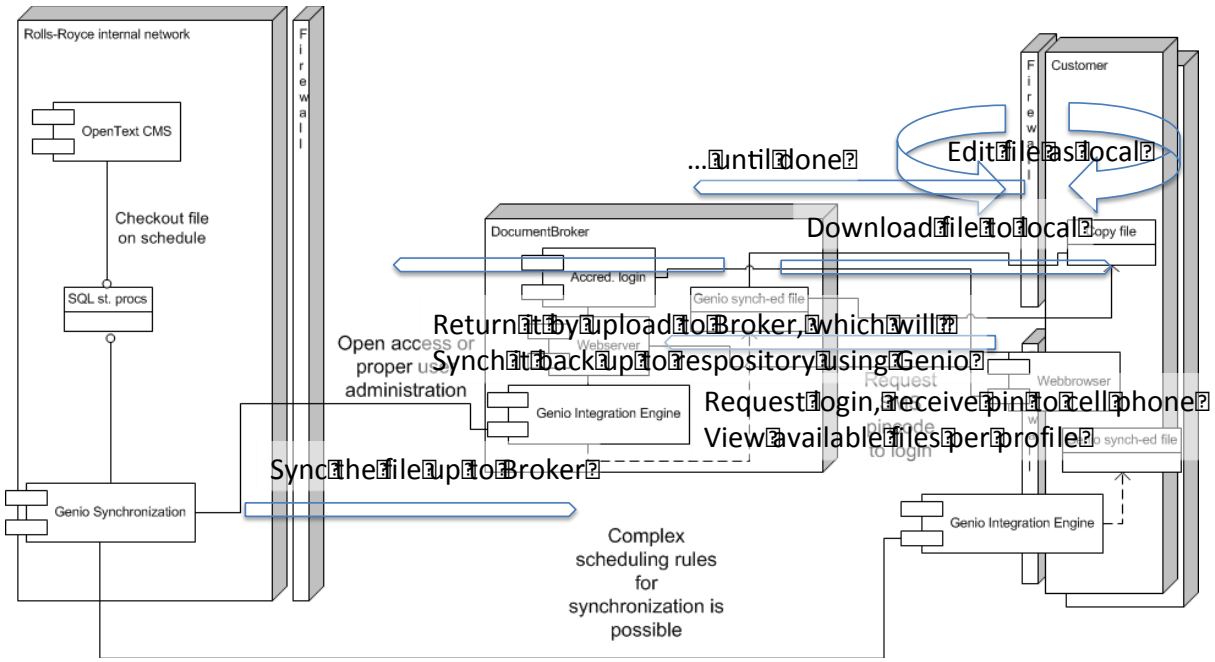

Figure 8: Full Open Text-based integration using "Genio"

7 SYNTERGY REPLICATOR

The feedback that we have gathered from Contesto representatives as well as the IT manager and engineers at Rolls-Royce themselves, however, indicate that the Genio solution is not sufficiently well developed and promoted by Open Text and therefore, another similar solution has been considered sourced, from a company called Syntergy. The history of their product, the *Replicator*, explain how come it has a tighter integration with Open Text than its competitors.

- Qualcomm, who were best known for its mobile phones, modems and the Eudora email software, started using a system called LiveLink in 1995, in order to control its document production. LiveLink was made by a small Northbrook-based company called Odesta Ltd.
- Odesta was acquired in 1995 by Open Text.
- Open Text started as a 1991 Uni project (Waterloo) to put Oxford English Dictionary on CD-rom
- Syntergy was formed in 1997, by a Qualcomm LiveLink superuser who turned consultant
- LiveLink was renamed to OpenText ECM Suite in 2008

The Replicator for Open Text ECM has got the following functionality listed:

- Multi-directional replication
- Proprietary transmission technology (based on Java APIs)
- Content mover
    - One-directional publishing, i.e.,
    - Export from CM repository to file/folder structure
- Bulk loader
    - One-directional reader
    - Import into CM repository from files
- LiveLink
- LiveMail
- Workflow

From the documentation and communication we have received, it seems that upon configuration, the Replicator mirrors a designated set of files from the Open Text CS, which is then made available through a browser client which can bridge between the Syntergy

replicator and Open text CS username identities, in order to furnish seamless access. The Replicator then uses Open Text APIs to keep the synchronized files up to date with the repository, and communicates directly with other instances to match their local files as well.

Our preliminary understanding of the architecture is documented in the figure below:

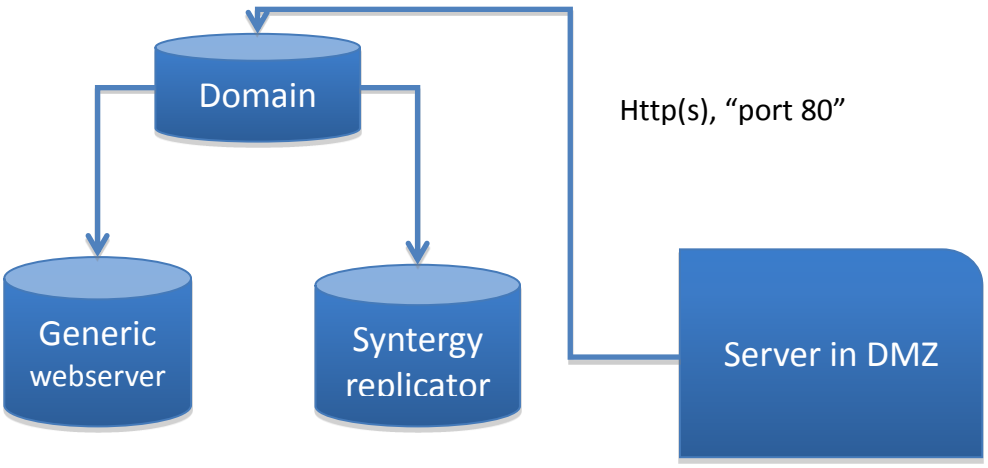

Figure 9: Outline of how we perceive the basic architecture of Syntergy replication

Syntergy implementation seems straightforward (judging from the whitepapers and information on their web pages).

- There seems to be a proxy implemented in Java (according to web seminar that have been published by Syntergy Ltd. on their website)
- Java file copy and remote authoring libraries/APIs are abundant, although we have not been able to establish if they have a proprietary version of it.
  - FTP
  - SMB
  - WebDAV (TomCat)
  - Commons FileUpload and FileUtils (Apache)
  - Com.oreillly.servlet.multipart,
  - Etc,

  are similarly easy to gain access to from the programming environment.
- Security may be handled with JNDI (Java Naming and Directory Interface) and SSL (Secure Socket Layer)
- *Diff* and *merge* algorithms are well-known
- Metadata are stored in CVS (attributes, location, type, etc)

A logical stream of commands in this set-up may be:

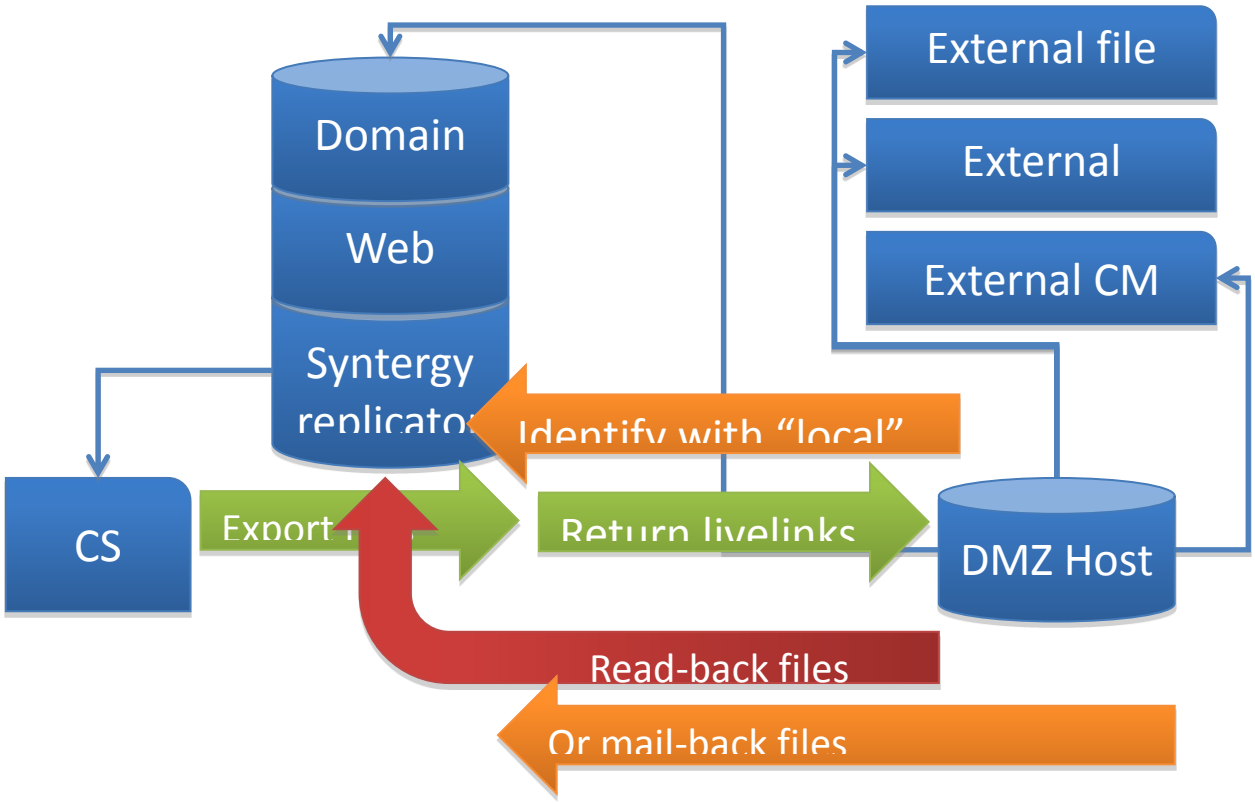

Figure 10: Flow of files/commands in the Syntergy setup

Syntergy represents a tightly integrated alternative, which on the other hand requires RRM to manage a service in the DMZ, including username and password appropriation (issue and or translation, plus the licensing costs). To some extent it also represents an even deeper lock-in with Open Text, since the solutions seem to end up, eventually, to be very highly interwoven.

The Syntergy components is running as a fully integrated service inside Open Text software, and offer necessary replication by being an administrative service for the Content Server.

On the other hand, Syntergy have got similar solutions for Microsoft SharePoint as well, which bodes well for tailoring to partners and customer who use that platform. Microsoft SharePoint is quite widely used in the maritime cluster of north-west Norway.

8 SIMPLY MOVING OPEN TEXT CS OUT INTO THE DMZ AND PROVIDING PARTNERSHIP-WIDE ACCESS TO IT

There exists, as the heading indicates, also a simple fourth alternative, whereby Rolls-Royce by allowing installation of licensed software in the DMZ at a dedicated server, i.e, moving the document hub one step closer to the enterprise network and costing it with the requirements to run Open Text Content Management Server for more users, can implement a simple platform for all users, internal and external alike. This may on the surface seem to represent an even more elegant solution than the Syntergy service, but there are several problems associated with it, the biggest concern from RRM being that files are thus having to be physically copied and will reside in two different repositories.

With the Synterghy replicator service, the files are still going to remain in the enterprise file server architecture, although the request-response service acting as a front-end towards the customer, may be located in the DMZ.

9 DISCUSSION

To summarize, we have presented four alternatives, of which one uses the eLink re-distribution mechanism of Open text, another that relies on rsync (or wrappers, such as Syncrify, Lip Sync or Delta Copy, to replicate file copies across an open Internet in a managed fashion so that only the difference between files is transferred and and that time stamps and file attributes may be maintained. In the latter instance, the traffic is encrypted, password protected and the services run with split privileges and minimum number of open port numbers.

Alternatively, one solicit more elaborate solutions from Open Text, such as their integration centre, or Genio. Finally, the existing Open Text instance may, just like web servers and email servers today, be hosted in the de-militarized zone, i.e., under RRM control, but accessible to customers and partners. It would require a different licensing scheme *and* extended (but not necessarily extensive) user management, but so would the solutions based on Syntergy as well. Syntergy comes across as relatively lightweight (in a positive way), however, since it is a service of the Content Server (CS), rather than standalone software.

In order to avoid software having to be installed or changed at the user site, one may replicate between the CS and a "hub", which we have called a document broker (DB). The non-interference with customer platforms does mean, however, that RRM must offer the service, almost obviously outside the intranet, in the DMZ or somewhere between, supported by a VPN (Virtual Private Network). This will in any case function as a web "sandbox", which encompasses all external user interaction to minimize the time and user/password distribution of access.

The login to the hub and/or VPN access should probably be protected with a token/password generator, either in hardware or software. RSAs SecurID is the world leader in this kind of technology, and a natural starting point for exploration of alternatives[7]. Their web site provides several alternatives for all relevant platforms. Many Norwegian vendors offer their products, and we have not attempted to present or recommend among them.

The DB may be eliminated by using the eLink functionality of the Open Text CS to receive files as email attachments, which are addressed and stored intermittedly in a dedicated folder from where it is uploaded to the CS' version management system.

If the eLink enabling is not used, the workflow functionality in Open Text needs to be implemented to extract and return files from the internal repository.

The technically most flexible solutions, i.e., those that are not "packaged (like Genio, Syntergy or a CS in the DMZ), comprises and configures the following elements, following through the information flow from the inside-out:

- An interface to the Open Text content management server system.

[7] http://www.emc.com/security/rsa-securid.htm

- Optional (see appendix): Adding an invisible watermark to each document, to make sure that its origin may be tracked back.
- File synchronization software solution (usually based on some variant of *rsync,* which may replicate a selection of files including their folder structures.)
- Files ought to be picked by profile, so that different users may gain access to different files.
- An Apache-based web server, which simply replicates the views from the synchronization engine.
- An SMS gateway, to make it possible for the server to request a mobile phone number-based registration[8] and validation through the ad-hoc generation a pin code (see appendix).

The following table summarizes the requirements analysis, and shows that there is good fit between the solution and the requirements for three of the alternatives:

	Replicated to customer site	*Replicated to DMZ*	*Within network: Non-replicated*	*All within DMZ: Non-Replicated*
Syntergy Replicator	Prohibitive overhead: Will need to ask customer or partner to install new licensed software	Low overhead: Runs as an integrated service component inside the Open Text instance of the DMZ, however, it elimitaes the need to physically duplicate the files pre-download by the customer. Little risk: Username and password for syntergy mirror may be different form Open Text. Quite efficient: May replicate only what is required. Robust: May work	N/A: Would require open access to enterprise network through firewall	N/A: Not much point in replicating if data is already accessible from DMZ

[8] http://www.kannel.org

		as additional backup. Quite flexible: May give "livelinks" limited scope and TTL (time-to-live). Partly exploits CS functionality. Very secure: May use Secure ID Software Token to authenticate for VPN without individual password.		
Open Text Content Server (CS)		Low overhead: Need replication software. Bigger risk: Need to assign local username and password for customer/partner, BUT may also use token-based VPN access. Fully expoits CS functionality, which also means that the files will be physically available in the DMZ.		Simple: Content directly available to all who have username/password. Efficient: No additional processing. Quite robust, but need separate backup routine. Very flexible: Have integrated Workflow. Fully exploits CS functionality, but veru vulnerable since the files only reside in DMZ.
SharePoint		May be more programmable, less expensive and more widely used among partners than any of the		N/A, since data is already in the DMZ in original form, so identical to solution that keeps CS in DMZ

		alternatives. The web interfaces of all are probably equally intuitive, though.		only.
Rsync	Low overhead: Customer/partner will easily be able to set up a repository to receive files	Very low overhead: Customer/partner can download files from a web-server in the DMZ without username and password.		
	Simple: Selected files may be pushed at fixed or scripted intervals. Very secure: Only the sshd and rsynchd ports need to be *outbound* open from the enterprise to the receiver. Competiencies in Unix-utilies, or similar, will be required. The set-up is highly tailored and therefore vulnerable.			

Table 1: Summary of requirements' analysis

The conclusion thus, on the background of having reformulated specifications and requirements slightly from the beginning, during this project, is either

- To move the entire Open Text Context Mangement System (CS) with a web interface into the DMZ (or require a separate cloud architecture hosted solution), and then base the entire solution on Content Server — The actual repository may probably be placed on a host away from the DMZ, with firewall-controlled access to it from the DMZ; *or*
- To replicate selected parts of the enterprise network-hosted CS-repository into the DMZ using Syntergy,
  and for both these first alternatives assign username passwords necessary to access by partners/customers, and/or RSA SecurID token-based VPN access with dedicated software on the client (new passcode generated every 60 seconds on the basis of initial token); or
- Push-replicate content from CS using workflow + *rsync* directly to a designated folder with the customer/partners network, from which they may edit and *eLink* it back using email *or* upload it to rsync, which mirrors it back.

The CS Management services should not be available from within the DMZ, but from a separate management network, which is separate from the repository as well. Separation may be made with parallel networks or internal VPNs.

It would probably be wise to implement a simple user identity system for bookkeeping, to keep track of tokens and match actual with stereotypical user identities. By this, we mean that standard external user names, numbered sequentially with a series of allowed user names that cannot be re-used until its "lease" has been returned, needs to be matched in a a table against actual external user identities, in order for Rolls-Royce to be able to find out who, when and when a specific user has accessed a document.

In this alternative especially, but also generally for any enterprise system, the design of the firewall architecture becomes crucial. The main alternatives from our discussion are all placed into the illustration below:

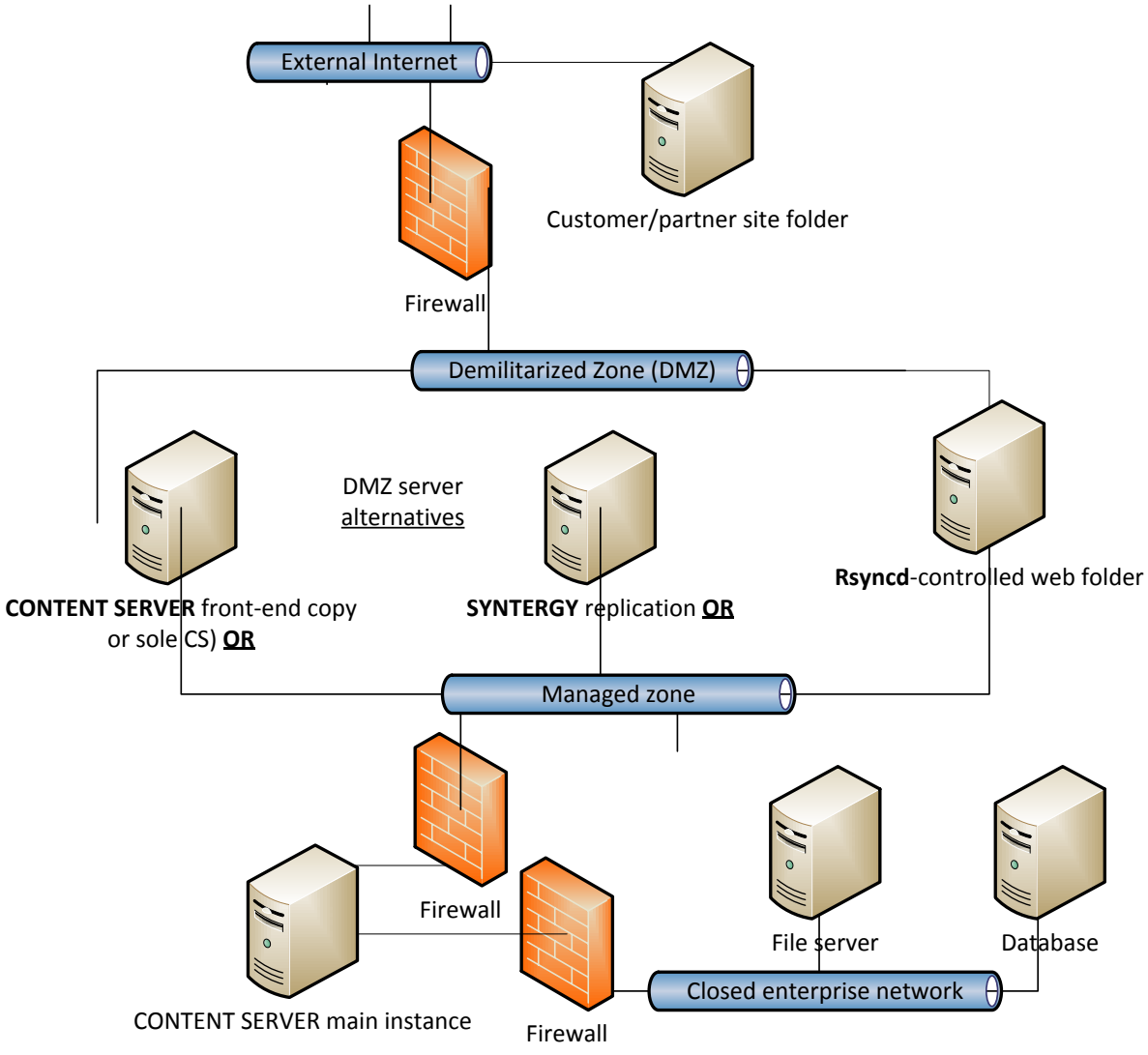

Figure 11: Firewall architecture for each of the three most viable alternatives

The suggestions thus made from this project are consistent with the mandate of creating a bridge between the Open Text context management system and an external repository, which is accessible from the outside of the intranet.

10 SUMMARY

We have in the current document described three real alternatives, plus one rather theoretical one represented by the unsupported Genio for Rolls-Royce document replication onto the DMZ. Our final recommendation is based on a narrowing down of the project mission, towards on-demand replicating (rather than duplicating) content to the DMZ. The following bullet points summarize our investigation:

- Figure out a good way to replicate information onto a server in the DMZ, which is:
  - Under Rolls-Royce management
  - Able to carry licensed software (unlike what should be required from customer/partner)
- Reduce administrative overhead and the amount of open port numbers
- Maintain workflow, version management and file integrity under Open Tex CS regime as much as possible
- The idea is fundamentally to separate sensitive enterprise network infrastructures from the Internet using a firewall architecture
- A DMZ (De-Militarized Zone) may be implemented according to various strictness and separation of concerns
- Services are put in the DMZ when they need to be shared with people on the outside
- Multiple firewall and virtual as well as physical network separation may be used.
- Example firewall setup
- Controlling servers in the DMZ
- A server in the DMZ might be single-purpose (for CS), and separated from enterprise networks by several firewalls
  - Web service in DMZ
  - Respositories are kept behind DMZ, but without the SNMP (Simple Network Management Protocol)
  - Management console, wich allows SNMP, with additional firewall between itself and the intranet
- Safe access to the services
- Should even protect the replication hub within the DMZ with a separate Network Access Server
- Requires a VPN connection to access the server

- VPN login could be based on NSA SecurID passcodes, which are sequenced by software or hardware generator per intial token on regular intervals (60 s)
- Rsync alternative:
  - No licensing of software: Use OS (Unix, Linux, Windows, OSX) utility to request or push files, which need to be extracted by *Workflow* directly to the customer/partners (C/P) file folders, or onto the “document hub”, which is in the DMZ.
  - C/P may down load with web browser, and return with email to Open Text using eLink, after working on the files.
  - Low overhead and cost, great security, needs internal resources.
  - Separates files from repository; break from version management and file control.
- Syntergy
  - Replicates only the necessary a selection of the Open Text repository
  - May separate selection on physical disk partitions, as well as make them ”read-only”
  - Translates between external and internal user names
  - May also push content onto MS SharePoint
  - Additional licensing costs, may have to explicitly integrate with Workflow (but possible)
  - Works as backup, no single point of failure
- Open Text CS throughout, placed in the DMZ
  - With sufficient security for document hub in DMZ,
    - *Management and data separation within DMZ*
    - *NSA SecureID pass code generation based on initial token*

it might be feasible to move the one, authoritative CS repository out into the DSZ for equal access by external as well as internal users, to achive:

- One single system, with a few more username/passwords to maintain
- Tight coupling with C/P, who accesses Open Text via its web interface in the DMZ
- Keeping within the workflow and version management system that has already been implemented.

However, this would leave the entire repository rather than the synchornization service for a selction of the entities in the respository residing outside the firewall.

Based on this analysis, the Syntergy replicator solution seems to have al the advantages of a monolithic, and thus consistent, CS implementation in the DMZ, but with added security inasmuch as only designated files might become available to intruders if security is compromised. For the fully centralized-inside-DMZ CS solution, a breach might give intruders access to all the files.

We believe, regardless of the solution selected, that it will be necessary to revise and strengthen (or keep in place a good auditing system to maintain current quality) the DMZ architecture. RRM may have to purchase and distribute software agents or calculators to support NSA SecurID VPN accreditation to the document hub in the DMZ, since safe username and password management is too easily a security flaw to be allowed without strict policies.

There is clearly some overhead costs needing to be taken into account, which is what we see as being a necessary cost of flexibility and loose coupling in the supply chain. We have not attempted to calculate this cost in the current project, but see it as a natural starting point for follow-up discussion when we conclude the current effort.

11 REFERENCES

[1] M. Vierhauser, P. Grünbacher, A. Egyed, R. Rabiser, and W. Heider, "Flexible and scalable consistency checking on product line variability models," presented at the Proceedings of the IEEE/ACM international conference on Automated software engineering, Antwerp, Belgium, 2010.

APPENDICES

Results of initial experimentation

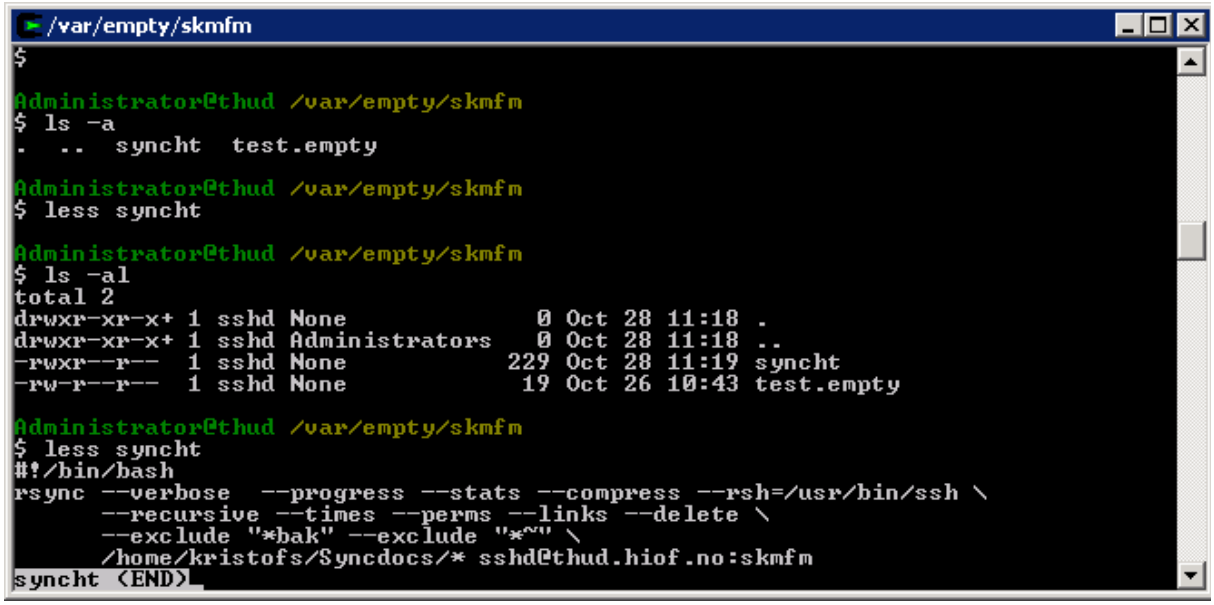

Figure 1: The terminal view of the test directory for user sshd at remote location

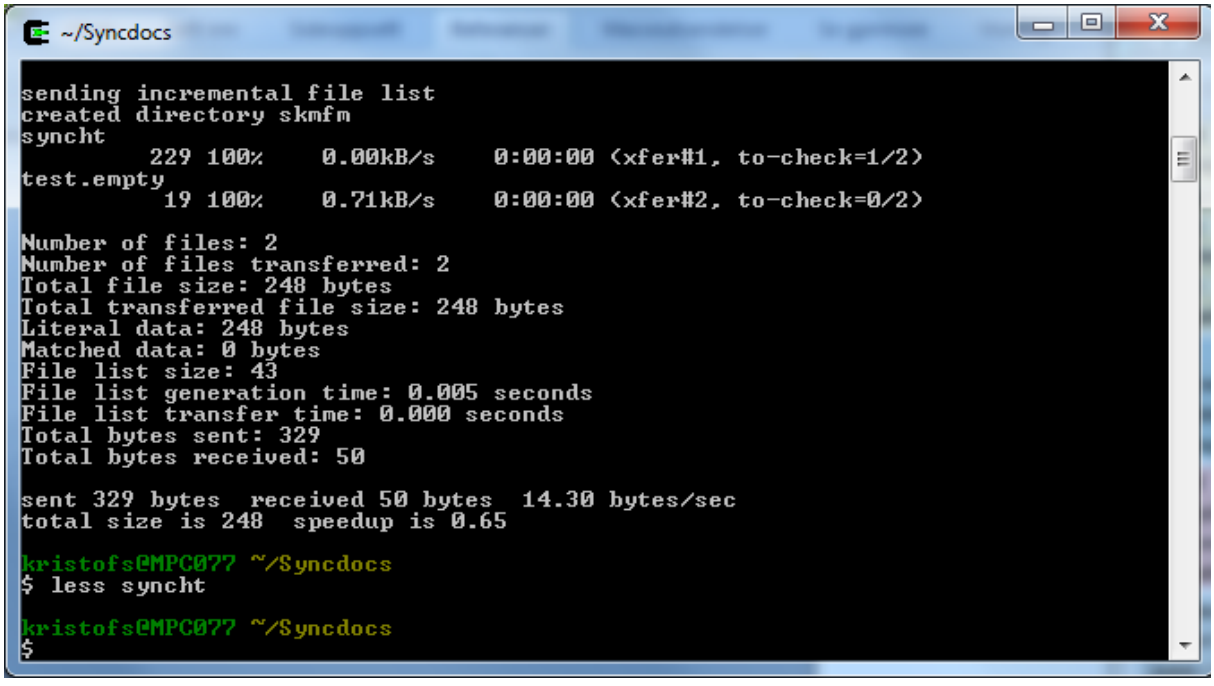

Figure 2: The terminal view of the local execution of rsync script

PUBLIKASJONER AV FORSKERE TILKNYTTET HØGSKOLEN I MOLDE OG MØREFORSKING MOLDE AS

www.himolde.no – www.mfm.no

2010 - 2012

Publikasjoner utgitt av høgskolen og Møreforsking kan kjøpes/lånes fra
Høgskolen i Molde, biblioteket, Postboks 2110, 6402 MOLDE.
Tlf.: 71 21 41 61, epost: biblioteket@himolde.no

NASJONAL / NORDISK PUBLISERING

Egen rapportserie

Kristoffersen, Steinar: *Safe and robust content distribution. Challenges and solutions related to Internet-based sharing of business critical documentation.* Rapport / Møreforsking Molde AS nr. 1211. Molde. Møreforsking Molde AS 50 s. Pris: 100,-

Oterhals, Oddmund; Bråthen, Svein og Husdal, Jan: *Diagnose for kystlogistikken i Midt-Norge – Forprosjekt.* Rapport / Møreforsking Molde AS nr. 1209. Molde. Møreforsking Molde AS 62 s. Pris: 100,-

Rekdal, Jens; Larsen, Odd I.; Steinsland, Christian og Zhang, Wei: *Eksempler på analyser av Køprising med TraMod_By. Konsekvenser av tidsdifferensierte bompengesatser i Oslo, Bergen og Trondheim.* Rapport / Møreforsking Molde AS nr. 1208. Molde. Møreforsking Molde AS.

Dugnas, Karolis og Oterhals, Oddmund: *Logistikkoptimalisering i Villa-gruppen. Kartlegging og forbedring av logistikkprosesser.* Rapport / Møreforsking Molde AS nr. 1207 KONFIDENSIELL. Molde. Møreforsking Molde AS. 53 s.

Rekdal, Jens; Larsen, Odd I.; Stensland, Christian, Zhang, Wei og Tom N. Hamre: *TraMod_By del 2. Delrapport 2: Eksempler på anvendelse.* Rapport / Møreforsking Molde AS nr. 1206. Molde. Møreforsking Molde AS. Pris: 150,-

Bråthen, Svein; Halpern, Nigel and Williams, George: *The Norwegian Air Transport Market in the Future. Some possible trends and scenarios.* Rapport / Møreforsking Molde AS nr. 1205. Molde: Møreforsking Molde AS. 82 s. Pris: 100,-

Hervik, Arild; Bræin, Lasse og Bergem, Bjørn G.: *Resultatmåling av brukerstyrt forskning 2010.* Rapport / Møreforsking Molde AS nr. 1204. Molde: Møreforsking Molde AS. 129 s. Pris: 150,-

Rekdal, Jens; Larsen, Odd I.; Løkketangen, Arne og Hamre, Tom N. (2012): *TraMod_By Del 1: Etablering av nytt modellsystem.* Rapport / Møreforsking Molde AS nr. 1203. Molde: Møreforsking Molde AS. 176 s. Pris: 200,-

Bråthen, Svein; Saeed, Naima; Sunde, Øyvind; Husdal, Jan; Jensen, Arne and Sorkina, Edith (2012): *Customer and Agent Initiated Intermodal Transport Chains.* Rapport / Møreforsking Molde AS nr. 1202. Molde: Møreforsking Molde AS. 153 s. Pris: 150,-

Bråthen, Svein; Draagen, Lars; Eriksen, Knut S.; Husdal; Jan, Kurtzhals, Joakim H. og Thune-Larsen, Harald (2012): *Mulige endringer i lufthavnstrukturen – samfunnsøkonomi og ruteopplegg.* Rapport / Møreforsking Molde AS nr. 1201. Molde: Møreforsking Molde AS. 125 s. Pris: 150,-

Kristoffersen, Steinar (2011): *Complete Documentation for Commissioning. Knowledge and document management in ship building.* Rapport / Møreforsking Molde AS nr. 1111. Molde: Møreforsking Molde AS. 32 s. Pris: 50,-

Hervik, Arild; Oterhals, Oddmund; Bergem, Bjørn G. og Johannessen, Gøran (2011): *NCE Maritime klyngeanalyse 2011. Status for maritime næringer i Møre og Romsdal.* Rapport / Møreforsking Molde AS nr. 1110. Molde: Møreforsking Molde AS. 35 s. Pris: 50,-

Fillingsnes, Anne Berit; Sandøy, Marit og Ulvund, Ingeborg (2011): *Ny praksismodell i sykehjem. Rapport fra et samarbeidsprosjekt mellom Molde kommune, Kristiansund kommune og Høgskolen i Molde.* Rapport / Møreforsking Molde AS nr. 1109. Molde: Møreforsking Molde AS. 50 s. Pris: 100,-

Oterhals, Oddmund; Johannessen, Gøran og Hervik, Arild (2011): *STX OSV. Ringvirkninger av verftsvirksomheten i Norge.* Rapport / Møreforsking Molde AS nr. 1108. Molde: Møreforsking Molde AS. 28 s. Pris: 50,-

Hjelle, Harald M. og Bø, Ola (2011): *Implementering av IT-systemer i verdikjeden for frossen fisk. Sluttrapport for FIESTA-prosjektet* . Rapport / Møreforsking Molde AS nr. 1107. Molde: Møreforsking Molde AS. 124 s. Pris: 150,-

Rekdal, Jens (2011): *Konsekvensutredning; Måseide – Vedde – Gåseid. Delrapport: Trafikkanalyse og samfunnsøkonomisk kalkyle for "Borgundfjordtunnelen".* Rapport / Møreforsking Molde AS nr. 1106. Molde: Møreforsking Molde AS. 112 s. Pris: 150,-

Hjelle, Harald M. og Bø, Ola (2011): *Sporbarhet, RFID og frossen fisk. Om potensialet til innføring av RFID-basert sporingsteknologi i forsyningskjeden for frossen fisk.* Rapport / Møreforsking Molde AS nr. 1105. Molde: Møreforsking Molde AS. 51 s. Pris: 100,-

Sandsmark, Maria og Hervik, Arild (2011): *Internasjonalisering av merkevarer i petroleumsnæringen i Midt-Norge.* Rapport / Møreforsking Molde AS nr. 1104. Molde: Møreforsking Molde AS. 41 s. Pris: 50,-

Bremnes, Helge; Hervik, Arild og Sandsmark, Maria (2011): *Merkevarer i petroleumsnæringen i Midt-Norge.* Rapport / Møreforsking Molde AS nr. 1103. Molde: Møreforsking Molde AS. 41 s. Pris: 50,-

Hervik, Arild; Bræin, Lasse og Bergem, Bjørn (2011): *Resultatmåling av brukerstyrt forskning 2009.* Rapport / Møreforsking Molde AS nr. 1102. Molde: Møreforsking Molde AS. 105,[42] s. Pris. 150.-

Oterhals, Oddmund (2011): *shipINSIDE – Vurdering av et nytt konsept for skipsinnredning.* Rapport / Møreforsking Molde AS nr. 1101. Molde: Møreforsking Molde AS. 25 s. Pris: 50,-

Hervik, Arild; Oterhals, Oddmund; Bjørn G. Bergem og Johannessen, Gøran (2010): *Status for maritime næringer i Møre og Romsdal 2010. Lysere ordresituasjon med utflating av aktivitetsnivået.* Rapport / Møreforsking Molde AS nr. 1011. Molde: Møreforsking Molde AS. 28 s. Pris: 50,-

Oterhals, Oddmund (2010): *Samseiling i Bodøregionen.Pilotprosjekt for utprøving av rederisamarbeid.* Rapport / Møreforsking Molde AS nr. 1010. Molde: Møreforsking Molde AS. 24 s. Pris: 50,-

Hjelle, Harald M. (2010): *FIESTA-skolen. Etterutdanning tett på egen verdikjede.* Rapport/Møreforsking Molde AS nr. 1009. Molde: Møreforsking Molde AS. 25, 91, [7] s. Pris: 150,-

Halpern, Nigel and Bråthen, Svein (2010): *Catalytic impact of airports in Norway.* Rapport/Møreforsking Molde AS nr. 1008. Molde: Møreforsking Molde AS. 112 s. Pris: 150,-

Oterhals, Oddmund; Hervik, Arild; Tobro, Roar og Bræin, Lasse (2010): *Markedskarakteristika og logistikkutfordringer ved offshore vindkraftutbygging.* Rapport/Møreforsking Molde AS nr. 1007. Molde: Møreforsking Molde AS. 35 s. Pris: 50,-

Merkert, Rico and Pagliari, Romano (Cranfield University); Odeck, James; Bråthen, Svein; Halpern, Nigel and Husdal, Jan (2010): *Benchmarking Avinor's Efficiency – a Prestudy.* Report / Møreforsking Molde AS no 1006. Molde: Møreforsking Molde AS. 74 p. Price: 100,-

Hervik, Arild; Bræin, Lasse og Bergem, Bjørn (2010): *Resultatmåling av brukerstyrt forskning 2008.* Rapport / Møreforsking Molde AS nr. 1005. Molde: Møreforsking Molde AS. 145 s. Pris: 150,-

Oterhals, Oddmund; Hervik, Arild; Øksenvåg, Jan Erik (Kontali Analyse) og Johannessen, Gøran (2010): *Verdiskaping og samspill i marine næringer på Nordmøre.* Rapport / Møreforsking Molde AS nr. 1004. Molde: Møreforsking Molde AS. 35 s. Pris: 50,-

Oterhals, Oddmund (2010): *Odim Abas. Verdikjedebeskrivelse og styringsmodell for prosjektgjennomføring.* Rapport / Møreforsking Molde AS nr. 1003. Molde: Møreforsking Molde AS. 38 s. KONFIDENSIELL.

Johannessen, Gøran; Hervik, Arild (2010): *Inntektsoverføringsmodell for lokale bil- og båtruter.* Rapport / Møreforsking Molde AS nr. 1002. Molde: Møreforsking Molde AS. 74 s. Pris: 100,-

Bråthen, Svein; Husdal, Jan (2010): *Fjerning av terreng hindre ved Kirkenes lufthavn, Høybuktmoen. Samfunnsøkonomisk analyse.* Rapport / Møreforsking Molde AS nr. 1001. Molde: Møreforsking Molde AS. 53 s. Pris: 100,-

ARBEIDSRAPPORTER / WORKING REPORTS

Rye, Mette (2012): *Merkostnad i privat sektor i sone 1a og 4a etter omlegging av differensiert arbeidsgiver-avgift. Estimat for 2012.* Arbeidsrapport / Møreforsking Molde AS nr. M 1201. Molde: Møreforsking Molde AS 19 s. Pris: 50,-

Bremnes, Helge; Kristoffersen, Steinar og Sandsmark, Maria (2011): *Evaluering av IKT-investeringer – et forprosjekt.* Arbeidsrapport / Møreforsking Molde AS nr. M 1103. Molde: Møreforsking Molde AS. 18 s. Pris: 50,-

Hervik, Arild; Hekland, Jon og Bræin, Lasse (2011): *Fiskeri- og havbruksnæringens forskningsfond (FHF). Screening av eksisterende erfaringer internasjonalt med måling/kartlegging av effekter av forskning innen fiskeri- og havbrukssektoren.* Arbeidsrapport / Møreforsking Molde AS nr. M 1102. Molde: Møreforsking Molde AS. 25 s. Pris: 50,-

Rye, Mette (2011): *Merkostnad i privat sektor i sone 1a og 4a etter omlegging av differensiert arbeidsgiver-avgift. Estimat for 2011.* Arbeidsrapport / Møreforsking Molde AS nr. M 1101. Molde: Møreforsking Molde AS. 17 s. Pris: 50,-

Dugnas, Karolis og Oterhals, Oddmund (2010): *Vareflyt og lageroptimalisering i Stokke AS.* Arbeidsrapport / Møreforsking Molde AS nr. M 1003. KONFIDENSIELL. Molde: Møreforsking Molde AS. 52 s.

Hervik, Arild og Bræin, Lasse (2010): *En empirisk tilnærming til kvantifisering av eksterne virkninger fra FoU-investeringer* Arbeidsrapport / Møreforsking Molde AS nr. M 1002. Molde: Møreforsking Molde AS. 59 s. Pris: 100,-

Bjarnar, Ove; Haugen, Kjetil; Hervik, Arild; Olstad, Asmund, Oterhals, Oddmund ; Risnes, Martin (2010): *Nyskaping og næringsutvikling i næringslivet i Møre og Romsdal. Sluttrapport.* Arbeidsrapport / Møreforsking Molde AS nr. M 1001. Molde: Møreforsking Molde AS. 15 s. Pris: 50,-

ARBEIDSNOTATER / WORKING PAPERS

Rønhovde. Lars Magne (2012) *Innovasjon i offentlig sektor : en studie av prosessene knyttet til initiering av og iverksetting av samhandlingsreformen i fem kommuner på Nordmøre*. Arbeidsnotat / Høgskolen i Molde, nr. 2012:1. Molde : Høgskolen i Molde. Pris: 50. –

Berg, Celia M.; Wallace, Anne Karin; Aarseth, Turid (2012) *IKT som hjelper og tidstyv i videregående skole : elevperspektiv på bruk av IKT i norsk og realfag*. Arbeidsnotat / Høgskolen i Molde, nr. 2012:2. Molde : Høgskolen i Molde. Pris: 100. -

Ludvigsen, Kristine, Jæger, Bjørn (2011) *Roller og rolleforventninger ved bruk av avatarer i en fjernundervisningskontekst*. Arbeidsnotat / Høgskolen i Molde, nr. 2011:1. Molde : Høgskolen i Molde. Pris: 50. –

Sandsmark, Maria (2011) *A system dynamic approach to competitive advantage : the petro-industry in Central Norway as a case study*. Arbeidsnotat / Høgskolen i Molde, nr. 2011:2. Molde : Høgskolen i Molde. Pris: 50. –

Bremnes, Helge; Bergem, Bjørn; Nesset, Erik (2011) *Coherence between policy formulation and implementation of public research support? : an examination of project selection mechanisms in the Norwegian Research Council*. Arbeidsnotat / Høgskolen i Molde, nr. 2011:3. Molde : Høgskolen i Molde. Pris: 50. –

Olstad, Asmund (2010) *Web-basert IT-system for beslutningsstøtte og kommunikasjon i operasjonell planlegging av prosjektorientert produksjon.* Arbeidsnotat / Høgskolen i Molde, nr. 2010:1. Molde : Høgskolen i Molde. Pris: 50.-

Bjarnar, Ove (2010) *Transformation of knowledge flow in globalising regional clusters.* Arbeidsnotat / Høgskolen i Molde, nr. 2010:2. Molde : Høgskolen i Molde. Pris: 50. -

Helgheim, Berit Irene; Jæger, Bjørn; Saeed, Naima (2010) *Technoøogical intermediaries as third part service providers in Global Supply Chains*. Arbeidsnotat / Høgskolen i Molde, nr. 2010:3. Molde: Høgskolen i Molde. Pris: 50.-

Angell, Truls; Jansson, Kjell (2010) *Will it be possible to achieve a simpler and efficient fare structure? – Case study Oslo*. Arbeidsnotat / Høgskolen i Molde, nr. 2010:4. Molde : Høgskolen i Molde. Pris: 50.-

Rekdal, Jens; Larsen, Odd I. (2010) *Underlagsmateriale for utredning av marginalkostnadsprising for tunge kjøretøy*. Arbeidsnotat / Høgskolen i Molde, nr. 2010:5. Molde: Høgskolen i Molde. Pris; 100.-

Bremnes, Helge; Sandsmark, Maria (2010) *An interdisciplinary study of competitive advantage.*Arbeidsnotat / Høgskolen i Molde, nr. 2010:6. Molde : Høgskolen i Molde. Pris: 50.-

Rapporter publisert av andre institusjoner

Vatnar, Solveig Karin Bø; Bjørkly, Stål (2011) *Forskningsbasert kunnskap om partnerdrap : en systematisk litteraturgjennomgang*. Rapport / Kompetansesenter for sikkerhets-, fengsels- og rettspsykiatri for Helseregion Sør-Øst, 2011-2. Oslo : Kompetansesenteret.

Lian, Jon Inge; Bråthen, Svein; Gjerdåker, Anne; Rønnevik, Joachim; Askildsen, Thorkel C.; Husdal, Jan (2010) *Samferdsel og regional utvikling : bistand til Nasjonal transportplan 2014-2023 : arbeidsgruppe for regional utvikling*. Rapport / Transportøkonomisk institutt, 1106/2010. Oslo : Transportøkonomisk institutt.

TFS 2012-08-22

MØREFORSKING
MOLDE

MØREFORSKING MOLDE AS
Britvegen 4, NO-6410 Molde

Telefon +47 71 21 40 00
Telefaks +47 71 21 42 99

mfm@himolde.no
www.mfm.no

Høgskolen i Molde
Vitenskapelig høgskole i logistikk

HØGSKOLEN I MOLDE
Postboks 2110, NO-6402 Molde

Telefon +47 71 21 40 00
Telefaks +47 71 21 41 00

post@himolde.no
www.himolde.no