Loughborough
University

This item was submitted to Loughborough's Institutional Repository (https://dspace.lboro.ac.uk/) by the author and is made available under the following Creative Commons Licence conditions.

For the full text of this licence, please go to:
http://creativecommons.org/licenses/by-nc-nd/2.5/

The Practicalities of Transferring Data between Project Collaboration systems used by the Construction Industry

S. Moses [a,*], A. El-Hamalawi [b], T.M. Hassan [b]

[a] *CICE (Centre for Innovative Construction Engineering) Civil and Building Engineering Department ,*
Loughborough University, Loughborough LE11 3TU, UK
[b] *Civil and Building Engineering Department, Loughborough University, UK*

Abstract

As part of an industry-lead initiative, standards have been developed to enable bulk exchange of project data between project collaboration systems used by the construction industry. Through the work of developing exchange standards to the practical implementation of data transfer, this paper examines the issues that need to be addressed by both solution providers and their clients when considering moving a project's data between two different instances of collaborative software. It highlights the practical difficulties of keeping the consistency of the data during the transfer process and gives solutions that can help to overcome these and other problems. The paper draws on experience gained through the development of vendor-neutral standards and real-life project migrations to put forward procedures which should be adopted by vendors, and gives insight into the underlying process for clients. The paper will form a vital framework for clients to determine if the benefits of moving projects between suppliers outweigh the difficulties associated with the move, and to help vendors put processes in place to best facilitate the transfer.

1. Introduction

Project Collaboration systems which encompass both project extranets and enterprise wide solutions are now widely deployed throughout the construction sector, with a wide number of vendors competing to supply these systems on either a global or per project basis [1]. These solutions allow for geographically dispersed project teams to work together, collectively increasing efficiency in the process and bringing greater profitability to their organisations [2,3]. Currently however project extranets are seen by many project participants as temporary repositories of information that are used for the duration of the project and then discarded. Clients also feel that once the project has begun they are stuck with the selected system for the duration without any reasonable way to transfer. Clients may desire to move their projects between different project collaboration systems for a number of reasons, or be forced into a transfer by events beyond their control. Examples of reasons clients may wish to move project data between systems with the potential benefits this could bring are as follows [4]:

- The client is unhappy with the service that they are receiving from the vendor and wish to move to another supplier who offers a more stable system and a better level of service.

- The vendor is unable or unwilling to continue the service that they are currently providing and the client needs to transfer project data to another supplier.

- When selecting a system, the client undertook pilot projects on a number of different systems to evaluate the products and wishes to move all the projects to the single selected system.

- The project on the extranet has been completed and the client wishes to bring the data in to their enterprise system for all employees to utilise the captured data.

To help address these client concerns with project collaboration systems a group of vendors formed the Network of Construction Collaboration Technology Providers (NCCTP), an association focused on developing standards that would aid data transfer between different systems and lead to a greater uptake of collaborative technology in the UK construction sector. Initially containing 6 collaborative vendors, the NCCTP has now grown to 9 members and is one of the projects managed by the British Construction Industry Research and Information Association (CIRIA). Participating vendor organisations produced an Extensible Markup Language (XML) based standard for the bulk exchange of project data between project collaboration systems [5], which was presented to industry at the Project Extranets V Conference in London [6]. The developed NCCTP Standard builds upon work done on the transfer of documents and associated metadata outlined in the DocLink Specification [7], and combines it with knowledge of how data is stored in a wide variety of collaborative systems. The DocLink Specification [8] extends the Industry Foundation Classes XML (ifcXML) model [9], with the NCCTP standard using the same class structure. The XML based bulk data exchange standard differs from previous prescriptive standards such as ISO [10] and [11], which attempted to standardise document metadata and construction specific document metadata, by providing an extensible model that can be mapped to each vendor's software.

Through the work done in association with the NCCTP Technical group, development and implementation of data transfer solutions at Causeway Technologies (a UK-based company) and actual industry deployments of the technology, this paper presents the practicalities associated with transferring project data between different extranet systems, and should be beneficial to both clients and solution providers. This paper shows clients what they should be expecting if they want to transfer projects between systems, from potential transfer duration to the difficulties that are involved. To vendors, the paper will show the areas that need to be considered in order to optimally accomplish the transfer of data with the least difficulties. The paper will also present some difficulties that may be encountered during the process.

2. Extracting Project Data

The first activity required in the transfer of project data between two different project collaboration systems is to extract all relevant data from the source system to an XML based external storage system. This extracted project's information must be self-contained with all the data required to recreate the project on the destination system included within the export. For example, all the users who have ever interacted with the project must be included even if their access to the project has since been removed. The partial good news for clients is since offline access to project data is necessary from a security perspective [12], most major vendors already have the ability to extract data for the purpose of project archives. It is questionable however whether or not these can meet the unique project transfer requirements, as each different system's unique export mechanism produces project data that is structured in a different way, and hence clients will be left with the responsibility for organising and paying for any transformation work required.

Equally for providers, even the actual extraction of their standard project data from the live environment, which potentially hosts numerous projects that are continually in use, is fraught with difficulties. It is of paramount importance when selecting both a time to conduct any project data export and the methodology utilised to get the data, to minimise the impact on the service level to other customers. It transpires that one of the advantages of project collaboration system use, with people able to work on the same data while been located anywhere throughout the world, is also a drawback when looking to extract the data. This 24/7 use of the service will mean that there are no natural points when the service is not in use or at very low levels of use, thus providers will need to monitor their usage before deciding when to conduct an export of data, Figure 1. Additionally the time required to complete the entire extract needs to be considered, with estimates as accurate as possible, while understanding that each project is unique. Projects that have been running for many years can amass large quantities of documents and associated metadata, and it is not uncommon to hold data in the order of tens of Gigabytes, and numbers of documents in the high tens of thousands. Extrapolating from the extractions documented in case studies included with this paper, projects of this scale could take many hours to export, adding increased strain to a live system.

Once a time for the export execution is agreed between the client and the provider, all the project participants need to be informed that data extraction is taking place. While some systems support the concept of marking projects as 'complete' and/or 'archived' effectively making it 'read-only', others do not. Those which do not have this functionality will need to deploy additional methods to ensure consistency of project

data. While it is technically feasible to set up export rules that will only export objects that have been created before a specified date, any changes to an object's metadata or any object deletions cannot be easily handled by these rules. For example if a document existed at the time that the export process was started, but was deleted by a user during the process, before the document was exported, then it would not be included in the exported project data. Therefore it is only possible to produce a snapshot of the project if it can be guaranteed that for the duration of the export process no user activity is undertaken on the project. Several different methods to achieve a genuine snapshot can be deployed, each with advantages and drawbacks:

- Inform all project participants that the project is been exported and that they should not enter it for the duration of the process. This option has the advantage that no changes need to be made to the project. However, it does rely on everybody obeying the request.

- Disable the accounts of all project participants for the duration of the export, restricting access to the system. This option ensures that no changes can be made to the project because users can not access the system. However if these users were collaborating on other projects then access to these would also be restricted; additionally extra effort would be required to both disable then enable the accounts.

- Restricting users' access to the project that is scheduled to be exported, allowing users to continue to use any other projects that they are members of on the system. Again the advantage lies in the fact that the project data cannot be changed by any of the project participants, but by adjusting the access controls the project is being altered anyway. Special care needs to

be taken if using the option to understand the impact it will have on the particular project collaboration system in question.

- Export the project from a restored backup of the system will ensure that no changes can be made to the project during extraction. This however will add additional time to the project transfer.

Ensuring that the project remains unaltered throughout the data extraction is also essential when it comes to verifying that the data has been exported, to ensure that it is an accurate representation of the project.

3. Data Consistency and Verification

The process of ensuring the extracted project data from the live system is an accurate representation of the project data held is a critical step in the transfer process. No matter how good the import application is, if the data supplied is faulty then the reproduced project will also contain errors. Clients need to be confident that data consistency will be maintained throughout the transfer, so a robust verification methodology is needed, but again a trade-off must be made between detail and time, with extra checks causing delays in the transfer process. While each utility used for extracting project data should/will come with its own error checking capabilities, these may need to be augmented with sanity checks to ensure that everything is as expected. The use of XML as the storage mechanism for project data outside of the collaborative tool makes these checks easier to perform.

To give prospective clients confidence that extracted data will be consistent and free from error, each vendor will need to have tested and verified the export routine in a large number of scenarios. Depending on the relationship that exists between client

and vendor, self-verification may or may not be adequate. To give the greatest confidence in data export, clients should seek some level of independent assurances that the extraction produces a true representation of the project.

Vendors seeking to support the ability of project data extraction should go through the following stages during the development and testing of their routine.

- Since the exported data will be held in XML format, manual checks of the exported data can be carried out on small test projects extracted from the system. Test projects should contain all the components of the source system, but in small quantities that can easily be checked.

- Automated checks of the exported system that compares the extracted information against that which is contained in the collaborative system, this process is substantially quicker then the manual checks. However, a full automated check of all the extracted project data is the equivalent of extracting the data once again from the system, meaning that this method is of the greatest use when testing extraction software due to time constraints and live system performance impact. This automated checking will provide confidence in the extractor so when it is used on large project in the live system this checking is not required.

- Importation of the data back into the system, will allow checks to be made on the exported data to ensure that it is capable of being imported back into the system correctly. Initial test should be conducted on small projects which will make the process of checking quicker while examining the maximum level of functionality of the extractor utility.

4. Data Security

As the data is extracted from the project collaboration system, all the security that had been applied to the objects and documents is lost, allowing anybody who gains access to the data-free reign to view or amend it. Therefore it is of paramount importance to add some level of security back as soon as possible, especially before any data is transferred from the secure environment from which it was extracted. Security is critical for two very different reasons; firstly to deny access to data and secondly to prevent the data from being changed while in transit. In any data transfer activity containing project data it must be clear whether or not the data has been modified by third parties while outside of the extranet systems. With the need for security firmly established, the vendor must only decide whether to encrypt the data during extraction, or after validation of the export. Should data be encrypted during extraction then additional time will be necessary to validate the exported data, adding to the total time required for the transfer. If security is added to the data as it is generated then the extraction process will take a longer period of time, furthermore verifying the extracted data to ensure that it is an accurate reproduction of that stored in the system will also take a longer period of time. Different methodologies for adding security are shown in table 1.

5. Data Storage
As project data is being transferred between different collaborative systems, some data will need to be stored outside of the system, regardless of whether it is being directly read and written or read, stored and then written. To enable data to be transferred between systems a common method of describing this data outside of the system is needed, to ensure that data is interpreted correctly by the destination system.

In recent years XML has emerged as the leading data exchange platform, and been used in a number of construction specific initiatives like ebXML [13], aecXML [14] and bcXML[15]. Figure 2 shows the data extraction process from the source system, then mapping and reproducing it correctly in the destination system via the XML based standard The methodology for the storage of project data should be selected from existing data exchange standards that both source and destination system support. Within the United Kingdom the majority of vendors support the NCCTP XML-based data exchange standard [16]. This attempt at the creation of a generic construction-based collaborative system model for individual vendors to map to is the latest in a number of proposed solutions [17], [18] and [19].

If the two systems do not support the same standard or support conflicting standards then agreement will need to be made on which standard should be used for the transfer. Should neither organisation support an existing standard then they are free to either develop a protocol for transfer between their two systems or both adopt one of the existing models. Constructing and implementing, or just implementing protocols for data exchange is not a quick process with many stages of testing required to ensure that the system works.

For any system used for storage of project data in a neutral method outside of the system the minimum requirements are:

- Support for User Accounts, associated user metadata and the overall organisational structure that they exist within.

- Support for objects, associated metadata and the overall system in which they are contained within the project.

- Access Control List methodology, allowing for object rights to be defined

- A method for recording audit events on the objects that are to be stored outside the system.

6. Physical Data Transfer between collaborative Systems

To allow for the project data to be imported into the destination system, it is necessary for the data to be physically moved from the source system to the destination system. To achieve this transfer several different options are available to the parties, including direct transfer over the internet and physical transportation via some external media, such as CD, DVD or Hard Disk Drive. Each of these options has its own practical difficulties and potential solutions with the selection been governed by the size of project data and the timescales involved.

Dependant upon network bandwidth, other requirements for this connection, it may actually be quicker to transfer data on a physical media between the two sites. However if transferring over a network then actions need to be made to minimise the risks associated with the transfer, with continual inspection of the process required to ensure that everything has transpired correctly. The data that needs to be transferred between the two systems will need to be secured and verifiable once it has reached the other terminal. Further difficulties and additional costs in achieving a speedy transfer between providers will also be caused by the time necessary to transfer the project data to the secure environments employed by many hosting organisations, which are required to meet IT management standards like ISO/IEC 17799:2005 [20] and ITIL [21].

7. Importing Project Data into a Collaborative System

If an extranet supplying company decides that it wishes to, or is asked by a client to import existing project data for either another vendor's system or another instance of its own environment, the general process can be split into two main sub-processes. Firstly the creation of all relevant user accounts and any associated information, and secondly the creating of the objects and their metadata. An overview of a standard sequence for the creation of project data is shown in figure 3.

It is important that the user information is created before the objects as object metadata directly relates to users that should exist on the system, for example objects are created by a particular user who exists on the system. Although this is the same for user accounts it still makes sense for users to be created first as they do not contain any references to objects.

When considering the import process it is important that the ability exists to import all of the data that was exported from the other system, while customers may wish to manipulate this data either prior to or during import to meet their own specific needs. The option to manipulate the data is critical to clients that in some instances will be paying for the project collaboration service by the seat, therefore to create a number of dead accounts just to maintain data consistency is not an economically viable proposal. The solution is to delete these users once the project import has completed successfully. However the exact strategy will need to be agreed between the client and project collaboration software supplier.

The key activity prior to import and indeed at the consultation stage is to manage the client's expectations of what will and will not be available in the new system. Although clients should be aware of what the newly selected system does, they will not be aware of the details of how it exactly works and what will not be supported in the new system. Therefore clients who seek to have their project data as close as possible to its previous taxonomy should seek to find a supplier who supports the same type of taxonomy model in their project collaboration system. For example support for the concept of nested folders, multiple files per revision or revision level permissions.

7.1 Importation of User Accounts

With the import of users there are lots of different things that need to be considered; firstly login names of each user must be distinct. Next the information provided about the user in the exporting system might not be consistent with information already existing in the importing system. Whether the vendor does the import manually or they set up rules to control how the users are imported automatically, vendors will need to make decisions depending upon the characteristics of the two systems and the number of users to be transferred. For small numbers of users the manual option is feasible, but as the number increases the chances of errors increase. To aid the automated procedure of user import a mapping file is either provided before import or during import, allowing the importer to match up those users that exist in the XML with those that currently exist in the system.

7.2 Importation of Objects

Importing the objects will require the actual list of mappings from the users and groups that were created on the system. When new objects are created in the system

they are assigned a new ID, because it is very unlikely that all of the IDs used in the XML will be available in the import system. Additionally different systems deploy different methods of giving objects ID, and thus it would be impossible to assign a string ID in a system that only supported integer IDs.

As each project collaborative system stores its data in different structures, the rules that govern the uniqueness of each object's ID need to be considered by the importer. It should therefore be assumed by the utility importing that object's ID are only unique within their own class, and that a folder for example could potentially have the same ID as a document object in the project. This point is of most importance to applications whose objects have IDs which are unique amongst a number of different classes, folder, document and alias for example.

The sequence of data export is also of key importance, as the majority of different extranet systems hold objects in a hierarchical structure. Therefore as new IDs are being assigned on creation it is impossible to create an object if its parent object has not already been created. Additionally the objects cannot be created directly to follow the structure as shortcuts, references to other objects in the project may refer to objects that have not already been created. If a Simple API for XML [22] processor is being used for reading the XML then a number of passes may be required in order to create the different types of objects. As an alternative to multiple passes through the entire XML document, the importer may elect to use the Document Object Model [23] allowing random access to the contents of the XML document.

As the objects are created in the new system, reference will need to be made to the user mapping data in order to tie together the metadata about the objects. The following items illustrate some of the common linkages between users and objects that need to be recreated as projects are moved between systems.

- The user who initially created the object.

- The user or group that owns the object, this potentially differs from the creator as object owners will have the responsibility of managing the object.

- The objects audit history which contains all actions users have performed on the object.

- The objects Access Control List (ACL), which holds information about the rights users, groups and organisations, have on the object.

8. Imported Data Consistency and Verification

As the physical transfer of project data between the two collaborative systems is completed, users once again will have access to the documents that it contains. To ensure continued confidence, the data that they are able to access must be comparable with their rights on the source system. Although an exact one to one mapping of a user's permission profile is highly unlikely between systems supplied by different vendors, checks need to be made to ensure that individual user's rights do not exceed those that were held previously. This is especially the case if automated mapping of permissions was undertaken at import time, rather then the removal of all permissions from all objects contained in the project.

When conducting checks there are two main methods that could be used to validate that the data is an accurate representation of what previously existed in the old

system. Direct comparison with the old system, via visual inspection of both systems, this is only possible if the system that exported the project is still active and that data browsing is still authorised. This type of checking is only of limited value as only small amounts of data can be checked in this manner, additionally since the new system data was generated solely from the XML and that the importer has no control over the XML that has been produced then this type of checking should only be used as a high level check. Since it is impossible to check every object that existed in the project to ensure that permission has been recreated to the satisfaction of client and no rights have been exceeded, clients must have confidence in the importers abilities.

Methods that can be employed to check that the data has been imported correctly are as follows:

- Compare the imported data against the data used in the import to ensure that it has been created correctly. The advantage of this is that any errors which exist would be reported and corrective action could be taken to repair the imported data. The disadvantage is that the amount of time required to do this check would be similar to the amount of time that it took to do the initial import. This check just like the initial import will impact on the performance of the system as it will be accessing the same live data as other system users. Additionally this will add extra time to the process and delay the project been active on the new system.

- Manual checks of a subset of the imported data to ensure the data has been imported as expected. The advantage of this method is that the checks are conducted with the minimum amount of impact on the live system, and relatively quickly.

- Client confidence in the import utility can be provided by the user doing a series of detailed testing to ensure that data from a variety of differing sources can be successfully imported correctly into the system. It is of paramount importance that the source system of the project data is known and testing has been conducted between the two systems to ensure that data can be moved successfully.

9. Case Studies

9.1 Transferring Project Data from a Project Extranet to an Enterprise Collaboration System

Since 2004 HBG, one of the UK's leading construction services organisations, have run an internal collaboration system based on Causeway's implementation of Livelink, for the construction industry. Prior to this, HBG had utilised a number of different hosted collaborative solutions to manage their construction projects. To improve information access for all HBG users to the information contained within these projects, they wished to have the data replicated in their internal system.

A suitable project was selected, table 2, from Causeway's hosted service, to be transferred to the HBG internal system. The migration was done utilising Causeway's Project Import Export Software which is based upon an extended version of the NCCTP's Bulk Data Exchange Standard. The extended version was selected for the migration since data has been transferred between instances of the same system.

The extraction of project data from the Causeway ASP system lasted 35 minutes and was done to coincide with minimum overall system utilisation. After completion of the export, project data was prepared, then transferred to Causeway's office from their

secure hosting environment, the whole process taking 1 hour. This extracted data, was stored in NCCTP bulk exchange format, including all the physical documents along with object metadata and user information, and was then encrypted to DVD and transported to HBG offices to be imported.

Once the data on the DVD had been unencrypted, the import of project data into HBG's environment was done in two stages. The first mapped the 115 user accounts from the Causeway ASP system to the existing user accounts on HBG's system, creating new accounts for those which did not exist, taking 45 minutes to complete. The second stage, lasting 75 minutes, imported the project's data into the HBG environment.

Review of the application logs and testing done by both Causeway and HBG showed that the project had been transferred successfully between the systems.

9.2 Transferring Projects from Pilot to Live Environments

Kier Group, a leading UK building and civil engineering contractor, first started piloting the use of an Enterprise Content Management System to manage their construction projects in 2004. After a number of successful pilot projects they decided to roll out the Causeway solution to their entire organisation, requiring them to deploy new hardware. However the initial pilot projects were still live and running on the pilot servers, and needed to be transferred to the new instance. Table 3 details the metrics of the projects migrated from the pilot server.

Kier and Causeway staff worked together using Causeway's NCCTP based project import export software to migrate each project, first to Kier's test environment, and then after successful validation of the process into Kier's live environment. As all project data was kept within Kier's network during the migration, there was no requirement to encrypt the data when transferring between servers.

The migration of each project between the pilot and live servers followed the same procedure; approximate times for each stage are shown in table 4.

1) Project data was exported from the pilot server and stored in an NCCTP based XML format.

2) Exported project data was validated, e.g. checking number of documents, revisions, etc.

3) Project data was moved from the pilot server to the live server.

4) User account mapping between the pilot and live server was done.

5) Project data was imported into the live server.

6) Imported project data was validated against the project on the pilot server.

The validation of exported and imported data and the migration utility log files showed that the projects had been successfully migrated between the pilot and live servers.

10. Conclusions

The size of the project data, the impact on overall system performance, and the need for that data to be available as quickly as possible will force the export of project data to be conducted at a certain time, and take a certain duration to complete. Only those small projects that can be quickly extracted and transferred speedily between the

systems could potentially be available by the next working day, thus allowing clients to experience the minimum of disruption. Larger projects that contain volumes of data that cannot be successfully transferred between successive days could potentially be transferred over a weekend or an extended holiday such is the case with most current software system upgrades or new installs. This approach would allow for risk to be reduced and time for any practical problems to be overcome before the data was required to be live again.

The project export procedure produces only a snapshot of the project at one particular time, and any information added to the project after this time will not be included in the data that will be installed on the destination system. Therefore it is of great importance that system users are made aware that the project is to be extracted at a certain time and that they will not be able to access the project on that system after that time.

To enable fast transfer of project data between different extranet systems any common storage methodology used, that has been created from a vendor neutral perspective, must be well supported and gone trough rigorous testing, giving clients confidence that transfer will be successful. However the very nature of a generic data store system means that it will not cover all the vendor specific attributes that exist within all different applications, and thus only those which have been found to be common can be transferred between the different applications. If however the transfer is being conducted between instances of the same vendor application then theoretically all data should be moved. These can be accomplished by a generic system utilising a method of extensibility in its data storage system.

The current work presented in this paper will help in facilitating future work in the areas of incremental project data transfer, which will address the practical issues identified with the bulk transfer of project data presented in this paper. This will eventually lead to a standard for real-time access to external repositories without the need to transfer project data between the collaborative systems.

11. Acknowledgements

The author wishes to thank the following providers of collaborative technology, 4Projects, Aconex, Asite, BIW Technologies, BuildOnline, Business Collaborator, Cadweb, Causeway and Sarcophagus who are participants in the NCCTP group and provided sample project data from their systems. The author also wishes to thank HBG and Kier Group for the data migration case studies included in this paper.

12. References

[1] Construction Industry Computing Association - Guidance on the Introduction and Use of Construction Extranets – Vendor Survey, Available from: http://www.cica.org.uk/pdfs/Vendor_Survey_Introduction_Use_of_Construction_Extranets.pdf, 2004 (Online).

[2] P. Wilkinson, Construction Collaboration Technologies – The Extranet Evolution, Taylor & Francis, UK (2005), 150-153.

[3] ITCBP. 2003, "Adopting internet based project collaboration software" Making IT work for your business, Case Study IT039, IT Construction Best Practice, Davis Langdon Consultancy, London, UK.

[4] S. Moses, T. Hassan and A. El-Hamalawi, The NCCTP Data Exchange Standards Imapct on the UK Construction Industries Collaborative Technology User Community, Proceedings of the 3rd International Conference on Innovation in Architecture, Engineering and Construction, Rotterdam,The Netherlands, 15 – 17th June 2005, 697-706

[5] The NCCTP Project Exchange Standard, Available from: http://www.ncctp.net, 2004 (Online), Date Accessed 13 December 2007

[6] Cole, T. (2004). The NCCTP Standard, Project Extranets 5 Conference, London, UK, November 2004

[7] Watson, A., & Davoodi, M., 2002, Transferring Project Document and Associated Metadata between company Document Management Systems and Project Extranets, eSmart and CISEMIC 2002.

[8] Leeds University, 2002, The DocLink Specification, http://www.engineering.leeds.ac.uk/civil/research/doclink/more.htm, Date Accessed 13 December 2007

[9] IAI, 2001, ifcXML recommended standard, Available from: http://www.iai-international.org/Model/IFC(ifcXML)Specs.html, Date Accessed 13 December 2007

[10] ISO, 2000a, Management Data (Metadata) for Technical Documents, IEC/ISO 82045, 2000

[11] ISO, 2000b, Technical Product Documentation – Metadata for Construction Documentation, PD ISO 19033, 2000.

[12] Construction IT Security Forum (CITSEC) 2004, IT Security in Collaborative Environments, http://www.citsec.com/it-security-in-collaborative - environments.htm, Date Accessed 14 May 2006

[13] ISO, 2001, Electronic business eXtensible markup language (ebXML), ISO15000, 2001

[14] IAI, 2004, aecXML Standard Documents, available from http://www.iai-na.org/aecxml/documents.php, date accessed 13 December 2007

[15] Lima, C., Stephens, J. and Böhms, M., 2003, The bcXML: supporting eCommerce and knowledge management in the construction industry, ITcon Vol. 8, Special Issue eWorl and eBussiness, 293-308, http://www.itcon.org/2003/22

[16] S. Moses, A. El-Hamalawi, T. Hassan, The Impact of the NCCTP Data Exchange Standard on providers of collaborative software to the UK Construction industry, Proceeding of the 2005 International Conference on Computing in Civil Engineering, Cancun, Mexico, ASCE, 2005

[17] Björk, B-C., Huovila, P., Hult, S., 1993, Integrated Construction Project Document Management (ICPDM), Proceedings of EuropIA '93, Delft, June 21-24, 135-146.

[18] Turk, Z., Björk, B-C., Johansson, C. and Svensson, K., 1994, Document Management Systems as an Essential Step Towards CIC. Proceedings of CIB W78 Workshop on Computer Integrated Construction, Technical Research Centre of Finland, Espoo, Finland, August 22-24.

[19] Rezgui, Y. and Cooper, G., 1998, A Proposed open infrastructure for construction project document sharing, Electronic Journal of Information Technology in Construction (ITcon). 3. 11-24.

[20] ISO, 2005, The International Security Standard, ISO/IEC 17799:2005

[21] ITIL, 2001, IT Infrastructure Library – The key to managing IT, Office of Government Commence, http://www-306.ibm.com/software/tivoli/resource-center/security/wp-itil.jsp , Date Accessed 13 December 2007

[22] SAX, 1998, Simple API for XML, XML-DEV Internet discussion group, http://www.saxproject.org/, Date Accessed 13 May 2006

[23] DOM, 1998, Document Object Model for XML, W3C Specification, http://www.w3.org/DOM/, Date Accessed 13 May 2006