

This item was submitted to [Loughborough's Research Repository](#) by the author.  
Items in Figshare are protected by copyright, with all rights reserved, unless otherwise indicated.

## Improving interoperability of AEC collaborative software through the creation of data exchange standards

PLEASE CITE THE PUBLISHED VERSION

PUBLISHER

© Scott Moses

PUBLISHER STATEMENT

This work is made available according to the conditions of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) licence. Full details of this licence are available at: <https://creativecommons.org/licenses/by-nc-nd/4.0/>

LICENCE

CC BY-NC-ND 4.0

REPOSITORY RECORD

Moses, Scott. 2019. "Improving Interoperability of AEC Collaborative Software Through the Creation of Data Exchange Standards". figshare. <https://hdl.handle.net/2134/34548>.



# University Library

Author/Filing Title ..... MOSES, SCOTT .....

Class Mark ..... T .....

**Please note that fines are charged on ALL  
overdue items.**

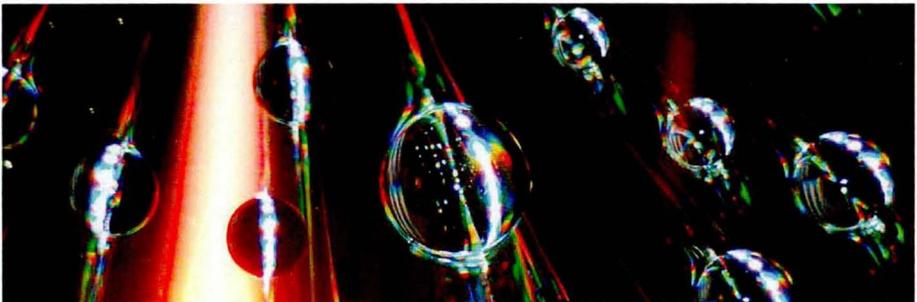
FOR REFERENCE ONLY		
--------------------	--	--

# Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

---

**Scott Moses**

---



---

Causeway Technologies  
Comino House  
Furlong Road  
Bourne End  
Buckinghamshire  
SL8 5AQ

---

Centre for Innovative Construction  
Engineering (CICE)  
Department of Civil & Building Engineering  
Loughborough University  
Loughborough  
Leics, LE11 3TU

0403604338





Loughborough  
University  
Pilkington Library

Date JULY '08

Class T

Acc  
No. 0403604338

---

# **IMPROVING INTEROPERABILITY OF AEC COLLABORATIVE SOFTWARE THROUGH THE CREATION OF DATA EXCHANGE STANDARDS**

By  
Scott Moses

A dissertation thesis submitted in partial fulfilment of the requirements for the award of  
the degree Doctor of Engineering (EngD), at Loughborough University

June 2008

© by Scott Moses 2008

Causeway Technologies  
Comino House  
Furlong Road  
Bourne End  
Buckinghamshire  
SL8 5AQ

Centre for Innovative Construction Engineering  
(CICE)  
Department of Civil & Building Engineering  
Loughborough University  
Loughborough  
Leics, LE11 3TU

## **ACKNOWLEDGEMENTS**

The research project undertaken throughout the four years would not have been possible without the support of a great many people, both formally through the mechanisms of the Engineering Doctorate and those who help informally. Firstly I wish to thank my two supervisors at the university Dr A El-Hamalawi and Dr T Hassan who provided much needed support throughout the project.

With the Engineering Doctorate based primarily in an industrial context I would like to thank members of my sponsor company Causeway Technologies, in particular Tim Cole who supported me throughout the project.

I would also like to acknowledge contribution to the project of each member organisation of the NCCTP with which I worked closely during the research.

---

## **ABSTRACT**

Today Collaborative systems are increasingly being used to manage project information on large and medium sized construction projects. The speed of expansion in use of these systems combined with the lack of consolidation has lead to a highly fragmented marketplace for collaborative products. Organisations participating in the construction lifecycle are currently free to select a collaborative system from any of the available providers, but once selected were unable to effectively change service provider until the conclusion of the project. This perceived lock-in along with concerns over the stability of some technology providers has created unease amongst the user community and is hindering the adoption of collaborative tools.

Since 2003 the bulk of major UK construction project collaborative software providers have been working together to develop standards that will allow for project data to be transferred between vendor applications. Under the umbrella of the Network of Construction Collaboration Technology Providers (NCCTP), a number of solutions have been designed allowing for project data to be transferred between heterogeneous collaborative systems.

Through extensive industry participation, this thesis shows how the theoretical work done in creating representations of collaborative systems can be applied to real world system to allow for data to be transfer in bulk, incrementally or in real time. The findings of work is presented in four peer reviewed papers, three technical reports and a number of supporting documents which comprise the developed data exchange

standards. Work in this field is continuing to evolve with the suppliers of collaborative systems seeking to implement additional integration.

**KEY WORDS**

Collaborative Software, XML, Standardisation, Construction Industry, Bulk, incremental and Real-time Data exchange.

---

## **PREFACE**

The research presented in this thesis is to satisfy the requirements of the Engineering Doctorate Programme (EngD), at the Centre for Innovative and Collaborative Engineering (CICE) based at Loughborough University, UK. This thesis represents work, undertaken in an industrial context, carried out between 2002 and 2007. The research programme was supervised by the CICE and jointly funded by the Engineering and Physical Sciences Research Council (EPSRC) and Causeway Technologies, a leading supplier of software solutions to the AEC sector.

The fundamental principle that underpins the EngD is that it must seek to solve one or more significant and challenging engineering problems within an industrial context. Differing from a standard PhD, it is better suited to the needs of industry, and provides a vocationally oriented doctorate in Engineering. Although part financed by the sponsoring company the resultant work must not only be of benefit to them, it must also benefit the wider industry.

The EngD is assessed on the basis of this thesis supported by publications, and or technical reports. This thesis is supported by one journal paper, two conference papers, and three technical reports. The papers and technical reports are numbered 1 through 6 and are located in the appendix of this thesis and should be read in conjunction with it when referenced.

## **USED ACRONYMS / ABBREVIATIONS**

API	Application Programming Interface
ASP	Application Service Provider
CICE	Centre for Innovative and Collaborative Engineering
CORBA	Common Object Request Broker Architecture
DCOM	Distributed Component Object Model
ECM	Enterprise Content Management
EDI	Electronic Data Interchange
EngD	Engineering Doctorate
EPSRC	Engineering and Physical Sciences Research Council
FTP	File Transfer Protocol
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol

---

ISO	International Organization for Standardization
NCCTP	Network of Construction Collaboration Technology Providers
OASIS	Organization for the Advancement of Structured Information Standards
SOAP	Simple Object Access Protocol
TCP/IP	Transmission Control Protocol / Internet Protocol
UN/CEFACT	United Nations Centre for Trade facilitation and Electronic Business
XML	Extensible Markup Language
XSLT	Extensible Stylesheet Language Transformations

## TABLE OF CONTENT

Acknowledgements.....	ii
Abstract.....	iii
Key Words .....	iv
Preface.....	v
Used Acronyms / Abbreviations.....	vi
Table of Content.....	viii
List of Figures.....	xi
List of Tables.....	xii
<b>1 Introduction.....</b>	<b>2</b>
1.1 Background to Research .....	2
1.1.1 Collaboration and Collaborative Technologies.....	2
1.1.2 Key Collaborative Issues facing the AEC industry.....	3
1.1.3 Current AEC use of Collaborative technologies.....	4
1.2 Context of the Research.....	5
1.2.1 Network of Construction Collaboration Technology Providers .....	5
1.3 Industrial Sponsor .....	6
1.4 Justification and Scope .....	7
1.4.1 Problem Definition.....	7
1.4.2 Scope of this research.....	9
1.5 Aim and Objectives.....	10
1.5.1 Aim.....	10
1.5.2 Objectives.....	10
1.6 Discourse Content Map.....	11
<b>2 Review of Existing and recent Work.....</b>	<b>14</b>
2.1 Background Research .....	14
2.1.1 Disruptive Technologies .....	14
2.1.2 Web services .....	15
2.2 Existing Collaboration Standardisation Proposals.....	16
2.2.1 DocLink Specification.....	16
2.2.2 ISO 82045 .....	17
2.2.3 ebXML (ISO 15000).....	18
2.2.4 PIX Protocol.....	18
2.2.5 JSR-000170 .....	19
2.2.6 ifcXML and aecXML.....	19
2.2.7 bcXML .....	20
2.2.8 eCoSpace.....	20
2.2.9 Semi Structured Data .....	21
2.2.10 Semantic Webs.....	21
<b>3 Methodology .....</b>	<b>22</b>
3.1 Introduction.....	22
3.1.1 Methodology Considerations .....	22
3.1.2 Methodology and Justification.....	24

3.2	Bulk Exchange of Project Data .....	26
3.3	Additional Application and Extension of NCCTP Bulk Exchange.....	28
3.3.1	Extending Bulk Exchange .....	29
3.3.2	NCCTP Based Project Data Archiving.....	30
3.4	Incremental Project Data Exchange .....	31
3.5	Real Time Project Data Exchange.....	33
3.6	Summary.....	35
<b>4</b>	<b>Research Activities and Findings .....</b>	<b>36</b>
4.1	Introduction .....	36
4.2	NCCTP Bulk Project Data Exchange.....	36
4.2.1	Clients' and Providers' Bulk Exchange Needs.....	36
4.2.2	Viability Assessment of Solutions.....	37
4.2.3	Industrial Workshops and Solution Design .....	38
4.2.4	Implementation of Bulk Data Exchange Standard with Causeway's Collaborative Solution .....	47
4.2.5	Practical Usage .....	49
4.2.6	Case Studies .....	50
4.3	Archiving Implementation of Bulk Data Exchange .....	54
4.3.1	Project Archiving Need .....	54
4.3.2	Viability Assessment of Solutions.....	55
4.3.3	Utilising Bulk Exchange Standard for Archiving.....	56
4.3.4	Implementation .....	57
4.3.5	Practical Usage .....	58
4.4	Incremental Project Data Exchange .....	59
4.4.1	The need for Incremental Data Exchange.....	59
4.4.2	Viability Assessment of Solutions.....	60
4.4.3	Industrial Workshops and Solution Design .....	61
4.4.4	Demonstration Project .....	64
4.4.5	Submission to NCCTP Group.....	64
4.5	Real-Time Project Data Exchange .....	65
4.5.1	Web Services Based API for Collaborative Systems .....	65
4.5.2	Real-Time Exchange Framework .....	73
4.6	Summary.....	76
<b>5</b>	<b>Conclusions and Implications.....</b>	<b>80</b>
5.1	Conclusions .....	80
5.2	Realisation of Aims and Objectives .....	80
5.3	Implications for the Sponsor .....	83
5.3.1	Business Exploration Opportunities .....	83
5.4	Implications for wider industry .....	84
5.5	Recommendations and Further Work.....	85
5.5.1	Academic Research.....	86
5.5.2	AEC Industry .....	86
5.5.3	IT Vendors .....	87
5.6	Critical Evaluation of the Research.....	88
<b>6</b>	<b>References .....</b>	<b>90</b>
<b>Appendix A The Impact of the NCCTP Data Exchange Standard on the Providers of Collaborative Software to the UK Construction Industry .....</b>		<b>96</b>

**Appendix B The NCCTP Data Exchange Standards Impact on the UK Construction Industries Collaborative Technology User Community .....108**

**Appendix C The Practicalities of Transferring Data between Project Collaboration systems used by the Construction Industry .....120**

**The Practicalities of Transferring Data between Project Collaboration systems used by the Construction Industry .....121**

**Appendix D An XML Based Standard to Enable Bulk Project Data Transfer Between Heterogeneous Systems .....138**

**Appendix E NCCTP BULK DATA EXCAHANGE STANDRAD .....152**

**Appendix F BULK EXPORT AND IMPORT CHECKLISTS .....183**

**Appendix G INDEPENDENT TESTING OF NCCTP IMPLEMENTATIONS204**

**Appendix H NCCTP INCREMENTAL DATA EXCHANGE PROPOSED STANDARD .....222**

**Appendix I NCCTP REAL TIME INTEGRATION SPECIFICATION .....340**

---

## LIST OF FIGURES

Figure 1 - Generic Structure of Research .....	23
Figure 2 - Interrelated Structure of EngD Project Components .....	25
Figure 3 - Key Activities and Deliverables for the Bulk Project Data Exchange Project .....	27
Figure 4 - Key Activities and Deliverables for the Extension of Core NCCTP Project Data Exchange Project .....	30
Figure 5 - Activities and deliverables of the incremental data updating project .....	32
Figure 6 - Key Stages and deliverables of the Real Time Integration Sub Project .....	34
Figure 7 - Opening a Text Document from Collaborative System using API's .....	71
Figure 8 - Saving Document Back to Repository using API's.....	71
Figure 9 - Browsing Collaborative Repository using API's .....	72
Figure 10 - Interacting with Document Created via API's in Standard Web Interface ..	72

## LIST OF TABLES

Table 1 - Discourse Content Map Chapter Breakdown .....	12
Table 2 - EngD Sub Project Key Deliverables.....	26
Table 3 - Incremental Updating Solution Evaluation Criteria .....	32
Table 4 - Schema Change Control Process .....	42



# **1 INTRODUCTION**

## **1.1 BACKGROUND TO RESEARCH**

### **1.1.1 COLLABORATION AND COLLABORATIVE TECHNOLOGIES**

Collaborative working lies at the heart of any successful team, and the way two or more participating individuals work together to jointly deliver the best solution to a common goal. This generic concept has been extended by Kalay (1999) to the AEC sector to become an agreement between cross organisational specialists to share their abilities to achieve the larger objectives of the project, as defined by the stakeholders. Indeed collaboration becomes more valuable to an organisation or organisations as more people are exposed to the same information, an extension of the Metcalfe's (1995) Law.

To enable effective collaboration between geographically and chronologically dispersed teams, collaborative technology was devised to create an environment where team members share and circulate ideas, experiences and knowledge. With the technology itself just an enabler allowing individual participants the ability to collaborate should they be willing to do so, it is people who collaborate, not systems or technologies (Butler Group 2003).

The term collaboration or collaborative working within the recent era for the construction sector was born out of the Latham (1994) Report's description of the communications which took place between partnering organisations. Collaborative working has been initially achieved using already available technologies such as phone, mail and fax progressing to email, ftp and groupware as they became available. It was

not until the explosion of web access in the late 1990's that the modern web-based construction collaboration applications started to become widely available.

### **1.1.2 KEY COLLABORATIVE ISSUES FACING THE AEC INDUSTRY**

Projects are on tight schedules and delays can cost large amounts of money to both client and contractor, making the availability of accurate current documents, drawings and specifications imperative to eventual success. Although the risk of the access to the collaborative environment being lost is relatively low, the consequences of this could be disastrous for the project. Even if it were possible to restore the project data into another collaborative environment, it would take a great deal of time. The challenge to collaborative providers in the construction sector is to deliver a quick and reliable mechanism of data transfer between different solutions.

At the completion of the project most collaborative providers will archive the data and present it to the client who they were contracted to provide the solution by, and then remove this project from their active collaborative environment. The result being the loss of valuable knowledge gained throughout the project by the other participating partners, and limits the ability to carry out a post project analysis and review (CITSEC 2004).

Due to the project orientated nature of the AEC sector, organisations of all sizes face the issue where their employees will be using multiple collaborative solutions, across multiple projects (Yeomans et. al, 2005). As users will be interacting with data through a number of different systems, they will not be able to easily access the advanced functionality which exists within each individual system, reducing the potential benefits

that these systems could have. Indeed employees are likely to need basic training on any new system they are expected to work with, incurring a financial implication for the project.

### **1.1.3 CURRENT AEC USE OF COLLABORATIVE TECHNOLOGIES**

The collaborative systems used by the AEC sector will primarily be focused on the areas of document management and version control, workflows, drawing management, the viewing and mark-up of drawings, searching and content security (Wilkinson, 2005). Known for their high service availability, collaborative systems offer uninterrupted access to data stored in the repository allowing effective project collaboration for teams located throughout the world.

Current deployments of collaborative solutions are varied depending upon the requirements of the project or organisation and have evolved rapidly since the initial in-house systems of the late 1990's, with three main types available, hosted extranets, hosted enterprises and enterprise solutions, Appendix D – Current Collaborative Systems.

Hosted extranet solutions are collaborative environments where project data is held in a remote secure location by the actual software vendor, with all infrastructure and applications managed by the vendor. These systems are usually used on a per project basis, although some bulk agreements exist especially with clients who do a large number of construction projects (4Projects & Tesco Express 2003).

Enterprise solutions are collaborative environments that are operated by the construction organisation, with the software provided by a collaborative vendor but managed by staff who report directly to the organisation. These systems allow for an unlimited number of projects that are more easily integrated into the company's other existing systems. Since the system belongs to the client it can be customised to their exact specifications which is not possible when using a single shared extranet solution e.g. ARUPLink (Cutler 2006).

Hosted enterprise solutions are collaborative environments that are operated by the collaborative software vendor for the sole use of a single client or a single project. The service is managed and maintained by the vendor similar to an extranet system, but since it is single use it can be customised according to the client or project's individual requirements. They differ from project extranets as they do not require 24/7 up times, only those which are required by the customer.

## **1.2 CONTEXT OF THE RESEARCH**

### **1.2.1 NETWORK OF CONSTRUCTION COLLABORATION TECHNOLOGY PROVIDERS**

The Network of Construction Collaboration Technology Providers (NCCTP) is a group of suppliers of collaborative software to the construction sector who operate within the UK marketplace

The primary aim of the network is to "promote the effective use of online technology to support collaborative working on projects and capital developments in UK construction.

It aims to increase interoperability between systems — engendering easy transfer of data through definition and adoption of standards. The Network will promote the benefits of using collaborative technology and demonstrate the value of collaborative working. Importantly, it provides a single independent body with whom clients can communicate regarding the future development of collaboration technology.” (NCCTP 2007)

To achieve these aims the structure of the NCCTP is split into three distinct tracks: technical, which works on the area of standardisation amongst the vendors, marketing, which promotes the use of collaborative technologies, and the steering committee which oversees the general direction of both groups.

The researcher’s primary responsibility within the NCCTP structure was to spearhead interoperability, working with the technical representatives of all the vendors to produce practical solutions. The role also included an examination of data exchange beyond the NCCTP’s initial bulk project exchange project to investigate the practicalities of extending the schema, project archiving, incremental exchange, and real time exchange.

### **1.3 INDUSTRIAL SPONSOR**

Causeway Technologies provided the industrial sponsorship and context for the undertaken research project. Causeway is one of the leading suppliers of software to the UK construction market with its solutions used throughout the world to facilitate the management of the entire construction process. Solutions provided include *Financial Management*, *Supply Chain Management* and *Enterprise Content Management*; the

latter of which includes the collaborative technologies which are an integral part of the research.

Causeway's construction specific ECM (Enterprise Content Management) solution can be delivered to clients as an ASP (Application Service Provider), hosted enterprise or enterprise, and is built upon the Livelink application provided by Open Text who is the world's largest supplier of ECM solutions spanning all industrial sectors.

## **1.4 JUSTIFICATION AND SCOPE**

### **1.4.1 PROBLEM DEFINITION**

The need for a mechanism to bulk transfer project data between different systems can be seen from both the perspective of client and vendor with each seeking different positive outcomes from the transfer ability. With vendors collectively keen to increase the utilisation of collaborative products, and clients seeking to gain these benefits in a secure and predictable environment. The following two paragraphs examine in more detail the need form each stakeholder group.

Through the implementation of a project data exchange standard the vendor community is collectively seeking to change some of the perceptions held by clients about the providers of collaborative tools. The standard is aimed at increasing confidence in the tools by providing a mechanism for the extraction of project data from one system and easy importation into another should the need arise.

Increasingly the larger clients have begun to select a single system to help to manage all of their projects in a single environment. These clients having spent large sums of money on systems and training staff on how to use these systems will require the ability to migrate their pre-existing projects from other systems to the one that they have purchased. They will be no longer willing to expend large amounts of money training staff on a new system on a project by project basis and will strive for their own system to be adopted for each project in which they participate.

In an attempt to meet the improvement targets demanded by Egan (1998) the UK construction sector has increasingly been employing project collaboration technology to manage the vast amount of information generated during a project. In the initial stages of this drive many construction organisations developed their own solutions, but as the complexity of solutions grew these were largely replaced by offerings from specialist software vendors. The move towards utilising a service provided by others raised a number of concerns about the reliability of the solutions and the business models operated by the vendors.

Construction organisations were concerned that once a particular service had been selected for a project, they were effectively locked in as it would be exceedingly difficult to get the data out of one system and into another. Additionally when the concept of enabling data transfer was first discussed many of the vendors of collaborative software were still largely unproven (Krojevski 2001). This coupled with industry publications questioning the financial predicament of some providers (Building 2004), continues to influence confidence in the collaboration provider industry. Client's

confidence in collaborative products has been further shaken by some early high profile failures of providers, and then their inability to access their data (Holden 2001).

Since the rapid growth of the sector has led to the creation of a large number of different suppliers of collaborative software solutions, industry watchers, such as Garner have been predicting widespread consolidation. This thinking has been reinforced by Lane (2003) who in his analysis of the collaboration industry expected that the UK collaborative software market would consolidate to around 3 main vendors. Those vendors leaving the marketplace would need to transfer their remaining active client projects to the vendors continuing to operate.

#### **1.4.2 SCOPE OF THIS RESEARCH**

The primary concern of this research is to deliver practical solutions which will enable the suppliers of collaborative solutions to the UK construction sector to more easily, quickly and seamlessly transfer project data between repositories.

The research initially focused on delivering a bulk exchange mechanism then moving onto incremental and real time proposals.

The research done worked with vendors who were largely primarily focused on delivering collaborative solutions to UK AEC companies, although the majority of these solutions are also used on projects throughout the world.

## **1.5 AIM AND OBJECTIVES**

### **1.5.1 AIM**

The project's primary aim was to increase the level of interoperability between the different vendors of collaborative software to the UK construction industry through standardised data transfer methodologies.

### **1.5.2 OBJECTIVES**

This aim was represented by a number of key objectives to be met throughout the project.

1. Provide and implement a mechanism which would enable bulk project data to be migrated between heterogeneous collaborative systems, which would be practical in real world situations.
2. Investigate the additional potential benefits that bulk project data extraction and bulk project data importation could yield, going onto proving these benefits were obtainable.
3. Propose a solution which would allow incremental data updates to be passed between different collaborative projects, replicating the content of one repository with another.
4. Propose a solution which would allow real-time integration between different collaborative systems participating on the same project, allowing the same content to be viewed and manipulated through multiple user interfaces.

## 1.6 DISCOURSE CONTENT MAP

Table 1, below outlines the discourse contents, indicating where sections refer to either parts of a paper, a paper in its entirety or a technical report located in the appendices of this thesis. Papers 1 and 2 are concerned with evaluating the proposed NCCTP XML Bulk Data exchange standard from the perspective of the vendor and client respectively. While paper 3 examines the practical issues which would be encountered by any clients and vendors during the transfer of project data between systems. Paper 4 highlights the business and technical need for transfer, how the NCCTP XML schema was developed, the common components of the generic collaborative system model created to enable transfers and the potential future work in this area. Technical report 1 outlines the NCCTP standard for bulk project data exchange, with reports 2 and 3 concerned with extending this to incremental and real time exchange respectively. Additionally the appendices F and G include a number of supporting documents written during the implementation phase of the project.

Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

Table 1 - Discourse Content Map Chapter Breakdown

Chapter 1	Background to Research Context of the Research Industrial Sponsor Aim and Objectives	Paper 1: Introduction Section Paper 2: Section 1 – Introduction Paper 4: Section 2 – Current Collaborative Systems
Chapter 2	Review of Existing Work General Subject Domain	Paper 1: Existing Standardised Data Transfer Methods Section Paper 2: Section 3 – Existing Standardised Data Transfer Methods Paper 3: Introduction Chapter Paper 4: Section 2 – Current Collaborative Systems
Chapter 3	Adopted Methodology Bulk Data Exchange Extending Bulk Exchange Incremental Data Exchange Real Time Data Exchange	Paper 1: Entire Paper Paper 2: Entire Paper Paper 3: Entire Paper Paper 4: Entire Paper
Chapter 4	Research Activities and Findings NCCTP Bulk Exchange Standard Proposed Incremental and Real Time Data Exchange Implementation within Causeway Practical Usage	Paper 1: The Industrial Need for Project Data Exchange Paper 2: Section 2 – The Industrial Need Paper 4: Section 4 – Generic Collaborative System Model Paper 1: Entire Paper Paper 2: Entire Paper Report 1: Entire Report Report 2: Entire Report Report 3: Entire Report
Chapter 5	Conclusion and Implications	Paper 1: Conclusions Section Paper 2: Section 6 – Conclusions and Future Developments Paper 4: Section 5 – Future Work Paper 4: Section 6 – Conclusions
Chapter 6	References and Papers Paper 1: Impact on Providers of NCCTP Paper 2: Impact on Clients of NCCTP Paper 3: Practicalities of Data Transfer Paper 4: XML Standard for Project Data Transfer Report 1: NCCTP Bulk Exchange Standard Report 2: Incremental Exchange Proposal Report 3: Real Time Exchange Proposal	



## **2 REVIEW OF EXISTING AND RECENT WORK**

### **2.1 BACKGROUND RESEARCH**

#### **2.1.1 DISRUPTIVE TECHNOLOGIES**

The term disruptive technologies was first coined by Clayton Christensen (Christensen, 1997), but it is not the case that these did not exist before this date (Access To Energy, 1998), (Integral, 2002a) and (Integral, 2002b). When defining the concept of a disruptive technology, Christensen showing that even though companies seem to do everything right, listening to customers, continue to develop technology to improve their existing product lines, and are always on the alert to what their competitors are doing, they occasionally lose major markets or major market shares, and they sometimes go out of business all together. However Christensen's definition of disruptive technologies as those which swoop in under a company's radar to offer low-end customers far better value has been questioned by Cohan (2000a) as to its relevance in the internet age. Cohan (2000b) highlighted in industry based examples by companies that have successfully managed disruptive technologies internally countering Christensen's theory, of how they should be handled

Since the original definition, there have been many attempts to redefine disruptive technologies in the light of what they mean to different businesses. However the original theory still remains relevant in the internet world as shown by the music, and film Industries' continuing battles with file sharing over the internet.

In the early 2000's web services were seen as a potential disruptive technology (Charlesworth, 2002), appearing on the research analyst Gartner's, Enterprise Application Integration Hype Cycle of new technologies (McCoy et al, 2003). They were seen as disruptive due to the potential they offered to change the way disparate computer applications communicated with each other. To test this, part of the EngD project examined the potential application which web services could have within the area of interoperability between different collaborative solutions offered to the UK AEC sector.

### **2.1.2 WEB SERVICES**

On the 12<sup>th</sup> July 2000, at the Microsoft Professional Development Conference in Orlando, Florida, Microsoft Chairman Bill Gates first unveiled the concept of 'Web Services' in the context as they are known today (Gates 2000). But while the words may have come from the mouth of the Microsoft's chairman, Web Services are not just a Microsoft creation. The technology was developed by many organisations which shared an interest in building electronic marketplaces. Although Gates reintroduced the modern concept of web services, their root can be traced back to 1975, when EDI heralded the launch of electronic data interchange (CMIS, 2003), as the first attempt to standardise business communication over a network. Since the launch of EDI other attempts have been made to standardise business communications over a network.

- CORBA (Common Object Request Broker Architecture)
- DCOM (Distributed Component Object Model)
- Unix Remote Procedure Call
- Java Remote Method Invocation

All these technologies are still in existence today, but to date they have failed to gain significant market share or enough momentum to succeed. It was the evolution in standardisation of the web which eventually made it possible to get all of the major vendors to agree on common transport protocols, HTTP, a universal business standard by 1997 running on TCP/IP, a mature standard when the web went main-stream in 1994. The final piece of the required puzzle fell in place with the eventual release of proposed SOAP 1.0 by Microsoft after over a years delay (Box, 2001).

Support for Web Services amongst the major IT vendors spread quickly and by the end of 2000, HP, IBM, Microsoft, Oracle and Sun had announced their commitment to Web services. (Hayward, 2002) shows that already some unexpected business opportunities have arisen, with MapPoint.Net, being used to combat credit card fraud in the retail sector.

## **2.2 EXISTING COLLABORATION STANDARDISATION PROPOSALS**

### **2.2.1 DOCLINK SPECIFICATION**

The Specification was produced as a direct response to the increasing number of organisations using project collaboration tools, by primarily Leeds University and collaborative software provider Sarcophagus (Watson & Davoodi 2002). It was aimed at aiding the movement of data between an organisations own internal repository and any of the many existing project collaboration solutions. DocLink proposed 25 generic XML transactions with which a corporate document management system could interact

directly with any compliant project collaboration system allowing the automated transfer of documents and associated metadata in either direction.

The DocLink specification defines a good system for transfer of information from one system to another but this is achieved by moving small amounts of data as responses to particular requests made by the other system. Since the requirement for the NCCTP Bulk data exchange would require for all project data to be transferred in one lump a methodology for storing data outside of the tools was required.

Additional problems with the DocLink Specification, if it were to be used for bulk data exchange was the requirement that the two systems be in direct communication with each other, exposing both systems to increased security risks. This coupled with perceived leaning to the Sarcophagus product meant that it would be difficult for other vendors to implement both from a commercial and practical view point.

### **2.2.2 ISO 82045**

This specification by the International Standards Organisation for document management systems (ISO 2000), describes the associated metadata which is attached to a document. The example Document Type Definitions (DTD) included with the current specifications do define elements that are common to the NCCTP Schema that has been produced, such as documents, and document revisions. However this is not directly comparable with the work that has been undertaken, as the NCCTP describes a wider range of objects. The similarities within the document elements are only a few as ISO 82045 goes into much more detail than that which is currently implemented by any

vendor and therefore not be suitable as the NCCTP's aim is to work with the existing structure of the systems, not impose a new structure.

### **2.2.3 EBXML (ISO 15000)**

Maintained by OASIS and UN/CEFACT and published as ISO 15000 this specification defines a whole set of standards that describe how organisations should conduct business over the internet; it describes a framework for conducting e-business using well defined XML messages within the context of standard business processes and partner agreements (ISO 2001).

The application of this framework, which is focused on continual communications, to the transfer of a collaborative project from one system to another, would be impractical as it would require a CCP (Collaborative Protocol Agreement) and both vendors to enter into a partnering agreement. However moving forward to the possibility of incremental and real time integration, where agreements such as these would be more likely the use of the ebXML framework for conducting eBusiness could be more useful.

### **2.2.4 PIX PROTOCOL**

The PIX (Project Information Exchange) protocol seeks to establish the best use of Information Technology for communication between the project participants, for each individual project, this includes selecting an appropriate collaboration system to manage document control (PIX 2004). It is recommended that the protocol be agreed as early as possible in the project so information can be shared amongst the participants, which works fine when the entire project team is in place at the start of the work. Issues could arise when decisions on systems are taken early on in the project and then imposed on

other participants as they join the team, leading to the situation where the same content can be stored in multiple repositories.

### **2.2.5 JSR-000170**

JSR-000170 is a Java based API for accessing data store in a content repository and collaborative systems, which outlines a series of methods which must be implemented by a vendor (JSR 2004). It defines the requests and responses but leaves the implementation to the individual vendor, which would allow real time access to data stored in the repository. The specification contains many of the generic functions required to interact with collaborative systems and formed an interesting reference point for the verification of the set of functionalities discussed by the NCCTP Technical Group. The JSR 170 Specification could not itself be used as vendors systems are written using different base technologies, and they do not necessarily have the java skills in house to implement this solution.

### **2.2.6 IFCXML AND AECXML**

With the wide spread emergence of XML as a data exchange platform, many new XML based standard approaches for the construction industry appeared for data exchange such as ifcXML (IAI 2001), aecXML (IAI 2004). While aecXML was aimed at producing a collection of transaction schemas, ifcXML did offer storage and transfer capabilities applicable to Collaborative systems. The Leeds University (2002) DocLink specification then extended the ifcXML model and applied this to the transfer of documents and associated metadata between collaborative systems.

### **2.2.7 BCXML**

Building and Construction eXtensible mark-up Language (bcXML) was designed to provide the European building and construction sector an XML based language to support the eBusiness communication between clients, architects, engineers, suppliers and contractors for the procurement of products, components and services (Lima et al, 2003). With its focus on eBusiness, the bcDictionary and bcTaxonomy defined in bcXML (Tolman et al, 2001) is well suited to defining the physical objects which exist in the AEC sector but not the virtual objects which exist in collaborative systems.

### **2.2.8 ECOSPACE**

The eProfessionals Collaboration Space (eCoSpace) Project is a European wide initiative started in 2006 aimed at delivering seamless, dynamic and creative collaboration across teams, organisations and communities through a personalised collaborative working environment by 2012 (ECOSPACE, 2006). This project is a three year project focused on four different areas of collaborative working, with the area of most interest to this thesis is their investigation into user-centric integration of collaborative tools. The aim being that shared workspaces become interoperable to avoid the deployment and learning of different solutions for being able to collaborate with people using different solutions (Prinz et al 2007). Participating in this project is one of the NCCTP member organisations, Business Collaborator, who will bring to the table valuable knowledge of interoperability gained during the standardisation efforts presented in this thesis.

### **2.2.9 SEMI STRUCTURED DATA**

The concept of Semi Structured Data precedes XML and refers to data which can neither be classified as structured, i.e. easily storable in a database or unstructured i.e. a text document. The work done by the NCCTP was aimed at creating a largely automated system which would allow for projects to be transferred between large numbers of different systems. This required the core data model developed to be ridged with element names and the general structure of elements clearly defined. However NCCTP does support the concepts of Semi Structured data through the extensibilities provided which allow for vendors to export custom data against defined NCCTP XML elements, they can included their own XML tags in an export.

### **2.2.10 SEMANTIC WEBS**

The Semantic Web is concept were the meaning of information available on the web is defined to such a degree that it is clearly understandable by both humans and machines (Lee et al. 2001). In such a scenario it would be possible for data, information and knowledge exchange between collaborative systems to be handled automatically since one collaborative application would be able to understand what the data in the other system was by simply viewing various web pages. The feasibility of this approach for use as a bulk data exchange methodology has a few fundamental floors which could not be overcome in the short term. One current implementations of collaborative software do not include enough semantic information in their web pages to allow for the true meaning of the data displayed to be understood, i.e. it would require substantial rework for every application. Secondly the physical size of some of the projects including documents would make transfer over the internet impractical.

### 3 METHODOLOGY

#### 3.1 INTRODUCTION

Research in the context of the Engineering Doctorate programme is a systematic investigation to discover, revise or extend theories that have practical application within the AEC Industry, and or extend general domain knowledge. The required systematic approach, or methodology, describes all the tools, techniques, methods and procedures to facilitate successful research. This chapter examines the different methodological approaches available and the rationale for the selection of appropriate methods for the undertaken research project, and sub projects, concluding with the overall research design.

##### 3.1.1 METHODOLOGY CONSIDERATIONS

Before it is possible to select an appropriate research methodology for the project it is important that a clear understanding of the different methodologies available is known. Dainty (2004) argues that all research methodologies can be categorised into one of three different types *Positivist*, *Interpretative* or *critical*. Creswell (2003) describes positivist methodologies as quantitative approaches that use positive observable phenomena to not only describe but also predict. Interpretative or phenomenalist methodologies are subjective and make no reliance on the postulates that underpin positivist methodologies. The content and pursuit is indicative of the researches intentions, and conclusions form qualitative narratives (Woods & Trexler 2001). Whereas critical methodology, which grew from perceived weaknesses in traditional methodologies (Yanchar et al. 2005), and is the evaluation how cultural, historical and

political factors have moulded experience creating subjective truths to be challenged. However whichever category of research methodology is selected the overall structure of the research, Figure 1, will follow the hourglass model (Dane 1990).

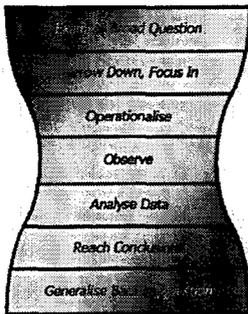


Figure 1 - Generic Structure of Research

Within the Engineering Doctorate Programme the broad area of research is decided up on by the University and the Sponsoring Company, with a suitable research engineer selected who is best able to conduct this research. The Research Engineer, in Collaboration with industrial and academic supervisors, then focuses the research, and selects how the goals are to be delivered.

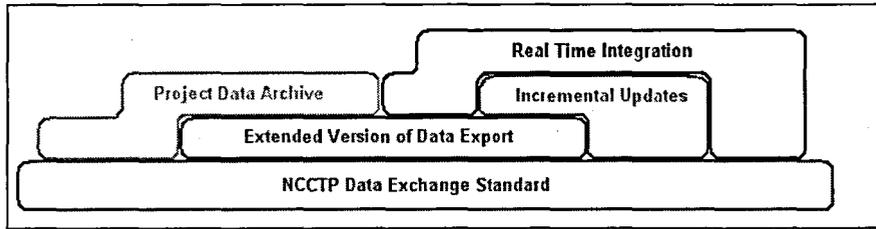
Once a suitable methodology has been selected, the research engineer can commence with the research, analyse the results and then draw suitable conclusions from the work. Finally the delivered research and its conclusions must be Generalised back to the initial Problem/Question that was set by the sponsor's.

Along with the different categories of research methodology specified earlier, there are a number of different research methods that can be undertaken including *Action, Cartography, Case Study, Classification, Content or Textual Analysis, Experience & Intuition, Experiments, Eye Tracking, Interviews, Mathematical Models, Participant Observation, Semiotics, Simulation, Statistical Analysis* and *Statistical Surveys* (Wikipedia, 2007). However not all these research methods are suitable for an engineering research project and only Action Research (Lewin 1946) and Case Study Research are deemed to be suitable (Yin 2002).

The Engineering Doctorate Programme allows for the research to take either the form of a single project that lasts the entire 4 year duration, or a series of sub projects that can be drawn together cohesively into a single project. Therefore in the situation where a series of projects have been undertaken a general methodology will be required for both the project in its entirety and each of the sub projects individually. The following section outlines the selected methodology for the entire project and those for the individual components of the project, stating the justification of these selections.

### **3.1.2 METHODOLOGY AND JUSTIFICATION**

The research that was undertaken to reach the aim of the overall project was conducted in five stages. These five phases of the project account for the research required for the entire project and that required for the individual sub projects. These separate components or sub projects were inter-dependent upon each other, forming a structure which is shown in Figure 2. Stage 1 set out to confirm the need for the research into data transfer between collaborative systems used by the UK AEC Industry, while stages 2 through 5 examined ways of best delivering the identified segments of integration to clients of collaborative products. Action and case study research methodologies were selected as the most appropriate to follow for the project as the research engineer would be embedded within the NCCTP technical group working with the vendors in workshops to collect sufficient information to design practical data exchange standards. These proposals would then be amended and validated through additional workshops conducted during and after implementation efforts by the vendors. The ultimate proof of the work success would be measured through practical application of data transfer between systems and presented as case studies.



**Figure 2 - Interrelated Structure of EngD Project Components**

Along with an overall methodology that controlled the project in its entirety each individual component had its own methodology which was focused upon delivering the best results for that phase. The individual methodology selections for the components are discussed in sections 3.2 through to 3.5, while the key deliverables for the project and sub projects is shown in Table 2.

**Table 2 - EngD Sub Project Key Deliverables**

<b>Sub Project</b>	<b>Key Deliverables</b>
Bulk Data Exchange	<ul style="list-style-type: none"> <li>• A Standardised method for the bulk Transfer of Project Data between different collaboration systems.</li> <li>• A solution Implemented by vendors in the UK AEC collaboration system supplier's marketplace.</li> <li>• Independent solution verification documentation.</li> </ul>
Extending Bulk Data Exchange and Project Data Archiving	<ul style="list-style-type: none"> <li>• Proof that the extensibility provided within the bulk data exchange schema can be used to support additional data stored within a project collaboration tool.</li> <li>• A Set of Generic Extensions which could be utilised by the NCCTP to increase the coverage of the bulk data exchange standard.</li> <li>• Additional Opportunities for the use of Extended Data Extraction and Importation.</li> <li>• NCCTP Export Based Project Archiving Solution</li> </ul>
Incremental Data Exchange	<ul style="list-style-type: none"> <li>• Proposed solution to the NCCTP which outlines a method of achieving incremental data exchange in a manor that complements existing NCCTP efforts.</li> </ul>
Real Time Data Exchange	<ul style="list-style-type: none"> <li>• A Proposed solution to the NCCTP Technical Group which outlines a methodology for the achievement of real time data exchange between collaborative systems used by the AEC sector.</li> </ul>

### **3.2 BULK EXCHANGE OF PROJECT DATA**

To satisfy the aim of producing a standardised mechanism for the bulk transfer of project data between heterogeneous collaborative systems and also to form a foundation for more advanced integration, any proposed solution needed to be as generic and widely adopted as possible. To increase the probability that the solution would be implemented by vendors in the construction collaboration supplier market, it was important that any proposal fitted within the existing implementations of software with

minimal changes required. Therefore a workshop approach was selected that would enable the technical architects of many major solution providers to be closely involved in the initial design and test implementation through to the final approval of the proposed bulk data exchange standard. These workshops, detailed in section 4.2.3, were conducted quarterly during the design phase to review progress and resolve any issues arising during the system modelling and implementation testing phases. Figure 3 illustrates the key activities flow and deliverables of this project phase showing the iteration process that the standard would be subjected to.

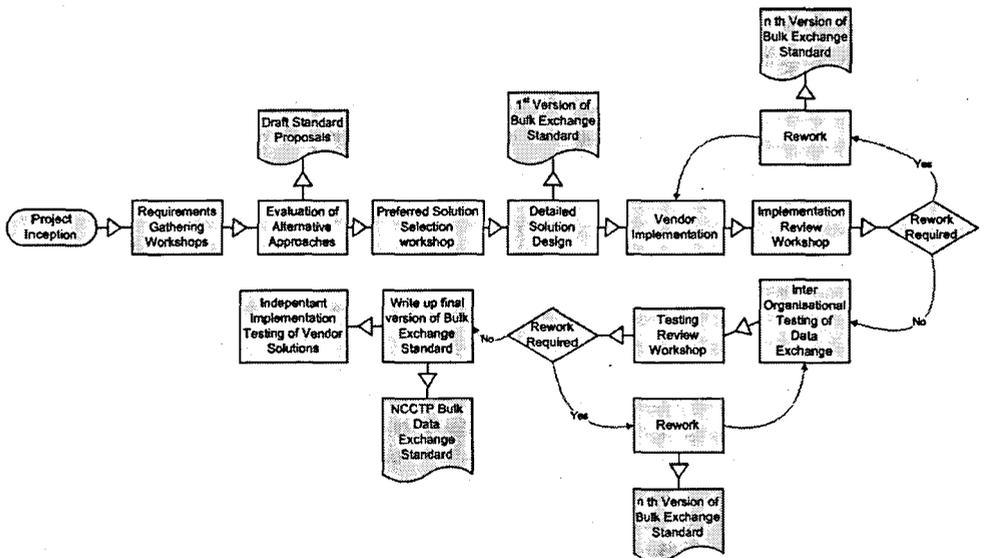


Figure 3 - Key Activities and Deliverables for the Bulk Project Data Exchange Project

The initial workshops aimed to establish the commonalities that existed between different collaborative systems through the examination of implementations of seven of the best known vendors in the UK marketplace. Allowing for a series of object class definitions and their inter-relationships to be defined, this could then be transformed into an object model defining a generic collaborative system used by the construction sector. An online discussion group was established in order for the model to be fine-

tuned through the implementation efforts of vendor organisations, with minor changes agreed online, while major changes being discussed and either accepted or rejected during the quarterly meetings.

Once the first agreed draft of the standard was available the participating organisation would then start to exchange sample project data to validate the object model that were made during the initial design of the standard with real project data extracted from other systems. Similar to the initial design phase of the model, changes would be made based upon the experiences of organisations as they tried to implement the solution, which would be ratified at the subsequent quarterly technical meetings.

Once intra-organisational testing of the data exchange standard was complete and each organisation was satisfied that it not only represented a model that their organisation could support but one that the wider industry could support the existence of the standard would be unveiled to the construction industry. To increase confidence in the data exchange standard a series of independent testing would be conducted by a 3<sup>rd</sup> party organisation to verify that vendors were capable of both exporting to and importing from the agreed standard.

### **3.3            ADDITIONAL APPLICATION AND EXTENSION OF                   NCCTP BULK EXCHANGE**

In this phase of the project an investigation was done into two possible application of the bulk data exchange standard.

### 3.3.1 EXTENDING BULK EXCHANGE

The designed NCCTP bulk data exchange standard defines will only those elements which are common amongst the majority of construction collaboration system, but includes built-in extensibility to allow individual vendors to export vendor specific information. To test this extensibility, Causeway's collaborative solution would have its NCCTP compliant project export expanded to export additional data in NCCTP format. This practical examination of the standards extension capabilities involved the following steps.

- 1) An examination of the additional data held within the Causeway collaborative solution which was not directly supported by the initial version of the NCCTP bulk data exchange standard.
- 2) Selection of a subset of available data, which would be the most appropriate to include as part of an exported project.
- 3) Design of structures/object models that would hold the additional information that could be contained within a valid NCCTP project export document.
- 4) Extending the export and import capabilities of Causeway's NCCTP utilities to deal with the additional project and object data.
- 5) Recommendations for using the extended export and import capabilities to meet other data exchange requirements.

Figure 4 shows the key activities and deliverables from this portion of the sub project aimed at using the extensibility incorporated into the NCCTP Bulk Data Exchange to include more project data in the export.

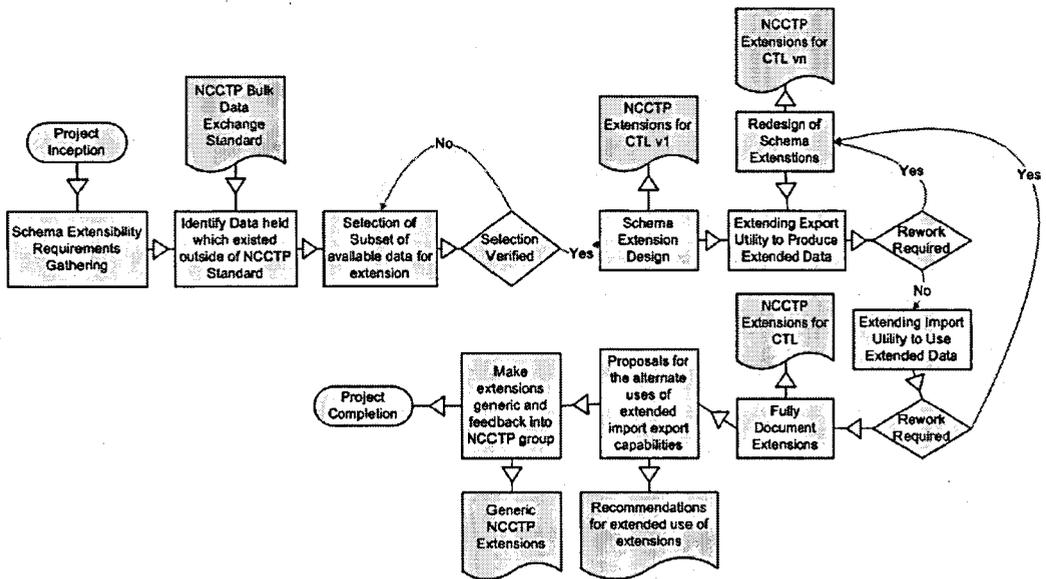


Figure 4 - Key Activities and Deliverables for the Extension of Core NCCTP Project Data Exchange Project

On completion of the research, the results of the extensibility analysis carried out on the NCCTP schema and any potential utilisation of the finding as separate product streams would be passed onto both the NCCTP technical committee and Causeway staff for evaluation and feedback.

### 3.3.2 NCCTP BASED PROJECT DATA ARCHIVING

The creation of a generic archiving solution which is capable of being applied to project data which has been created in a number of different collaborative solutions offers tangible benefits to both users and suppliers. This phase of the project will examine if an archiving solution could be applied to project data extracted in NCCTP format from any collaborative system. The sample projects used for this analysis would be those generated during the intra-organisational testing of the bulk project exchange standard with the aim being to give the impression they had all been generated from the same

system when viewed through the archive. The practical examination of NCCTP exported data usability for project archives involves the following primary tasks.

- 1) Review of the existing functionality offered by those vendors who do currently offer post project archiving functionality.
- 2) Selection of appropriate technology for the creation of the archive viewer.
- 3) Analysis of the suitability of the data structure of an exported project in NCCTP format to archive viewing.
- 4) Document the changes required to allow exported project data to be viewed in an archive viewer
- 5) Create Transformation application which will restructure data to the format required by the archive viewer.
- 6) Create the archive viewer.
- 7) Recommendations on the suitability of an archiving solution based on NCCTP extracted data.

On completion of the research into the creation of an archiving solution based on the data extracted in an NCCTP bulk project exchange export, the finding will be passed onto both the NCCTP technical committee and Causeway staff for evaluation and feedback.

### **3.4 INCREMENTAL PROJECT DATA EXCHANGE**

The incremental updating of project data between two collaborative systems used by the construction sector is a logical extension of the generic collaborative system object model, developed in the bulk data exchange of project data within this research.

Therefore the start of this work was constrained by the completion date of at least the first recognised version of that bulk exchange standard. Figure 5 shows the flow of this sub project, along with the key deliverables to be produced during its progression.

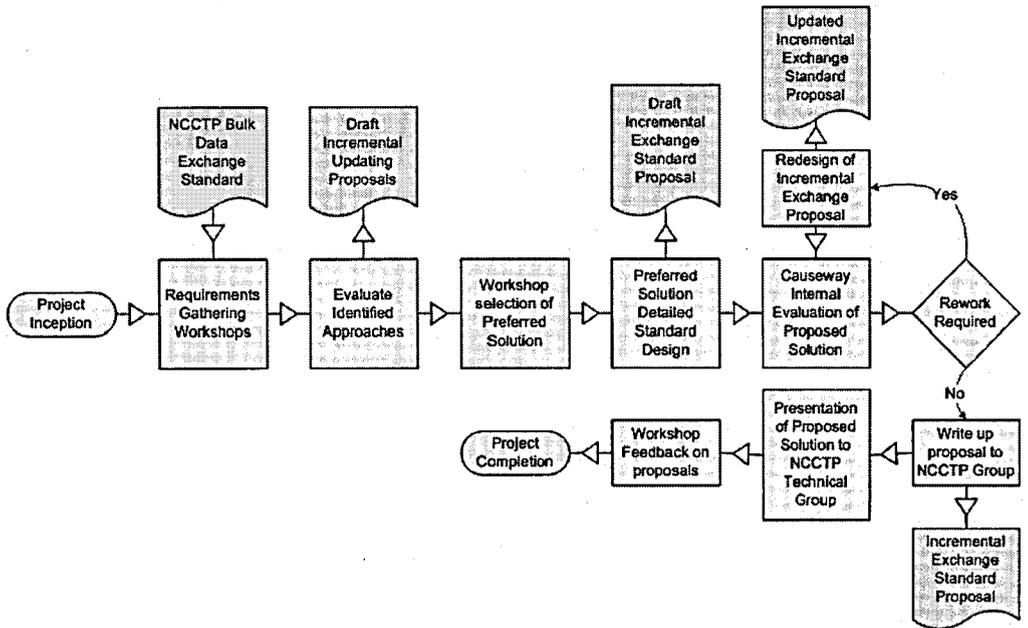


Figure 5 - Activities and deliverables of the incremental data updating project

The initial step in the project will be to establish the need for the creation of an incremental project updating solution amongst both the vendors and clients, and then the evaluation of all the available methodologies that could be adopted. These different approaches would then be debated with the other participating members of the NCCTP Technical committee, and the most promising selected for further investigation, based upon the evaluation criteria, detailed in Table 3.

Table 3 - Incremental Updating Solution Evaluation Criteria

Criteria	Description
Feasibility	Is the proposed solution practical in real world situations
Design Complexity (Cost)	The amount of time required to design the proposed solution

Implementation Effort	The amount of time required implementing the solution, including testing and enhancing the original design.
Solution Capabilities	A solution which maximises the data which can be incrementally updated, while reducing the complexity of the incremental updates.

The selected methodology to achieve the goal of incremental project updating would then be written up in to an NCCTP proposal document outlining the recommended approach to delivering an incremental project data exchange solution. This document would then be evaluated by the member organisations, as a standardised way of incrementally updating project data, and once the evaluation is complete this document would become the proposed NCCTP incremental project data exchange standard.

### **3.5 REAL TIME PROJECT DATA EXCHANGE**

The real time exchange or access to project data which is or was initially stored in a different system to the one through which the user is attempting to access it, has been the long time goal of the NCCTP technical committee since its inception in mid 2003. Although this project has been ultimately planned since the start it does rely upon the successful completion of the bulk data exchange work and to a lesser extent the incremental project data exchange work prior to its commencement. The following flowchart, Figure 6, outlines the key stages in which the real time project should progress through in order to produce an outcome that can be effectively incorporated and utilised by vendors and clients.

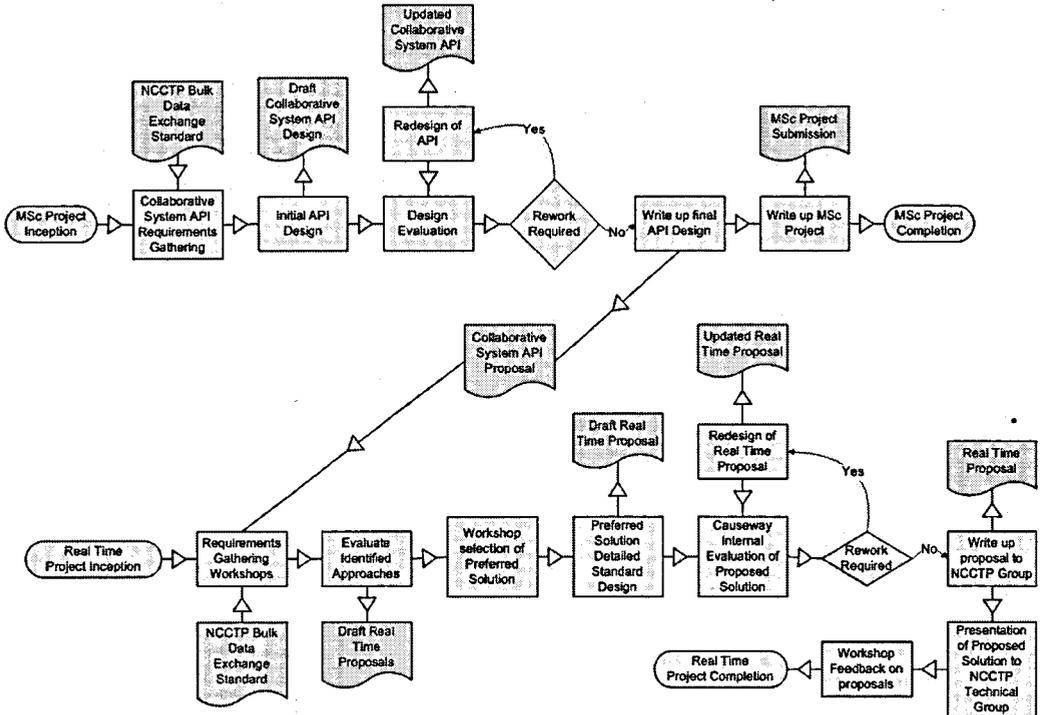


Figure 6 - Key Stages and deliverables of the Real Time Integration Sub Project

Since vendors are generally reluctant to undertake experimental work which may not be required by the user community, it is important to establish the need for and potential business models that real time integration between different vendors will open up. This can be achieved through the analysis of previous work in the sector and discussions with prominent clients who are in the position of utilising multiple solutions concurrently.

Once that the justification of the work has been clearly established and is accepted by the majority of the vendor community the next stage was the creation of a series of proposals which would allow for data to be available in real time through a variety of different user interfaces. These proposals will be based upon an evaluation of work done

previously in other NCCTP activities, and work carried out by other teams in both the construction and non construction sectors.

The NCCTP Technical committee have then evaluated the different options and selected the one that they wish to progress in detail, with this option then being expanded into a proposal that could be adopted by the NCCTP as a standard for real time integration between collaborative systems used by the AEC sector.

### **3.6 SUMMARY**

This chapter has discussed the methodologies adopted for the EngD project in its entirety, and those adopted for the individual sub-projects which combined together to form it. It provided an overview of the research methodologies available and a justification of those selected for particular components, to deliver the overall research design.

## **4 RESEARCH ACTIVITIES AND FINDINGS**

### **4.1 INTRODUCTION**

This chapter describes the research carried out to meet the aim and objectives of the entire EngD project, with each of the sub projects presented in its own section of the chapter.

### **4.2 NCCTP BULK PROJECT DATA EXCHANGE**

The initial stage in any EngD project is to examine the work that has already been done to ensure that a genuine real world problem exists, or that any existing solutions can be improved. Once the need had been established, and any existing solutions examined a process of industry workshops was undertaken to develop a generic solution which would then be implemented by participating vendor organisations. The following sections discuss each of the activities in more detail.

#### **4.2.1 CLIENTS' AND PROVIDERS' BULK EXCHANGE NEEDS**

At the start of this sub project a literature review was carried out to identify previous work in the following areas:

- The barriers to the adoption of project collaboration tools.
- The client's need for the ability to transfer project data between different tools.
- Any existing studies that had examined data exchange in a construction context, especially vendor neutral implementations.

- And; existing studies which had examined data exchange outside of a construction context.

The paper “potential impact the bulk data exchange standard may have on the user community”, Appendix B, contains an evaluation of the need for data transfer. It highlights the following, (explained in detail in the paper), as the client’s need for bulk project data transfer:

- Once started, clients are locked in to using the solution throughout the project.
- Confidence in the stability of the providers of collaborative solutions.
- Anticipated Consolidation of the sector.
- The increase in use by larger organisation of a single enterprise solution to manage multiple projects.
- Current requirement for organisations to use multiple project collaboration simultaneously.

The needs of vendor’s to create bulk project data exchange were primarily focused on increasing client confidence in the vendor marketplace, “The Impact of the NCCTP Data Exchange Standard on the Providers of Collaborative Software to the UK Construction Industry“, Appendix A.

#### **4.2.2 VIABILITY ASSESSMENT OF SOLUTIONS**

A solution which would allow for project data to be bulk extracted from one system and imported in a timely way into a number of other systems would need to address the following points:

- Since all project collaboration tools have their own unique underlying database schema, a common description of the structure would be needed. This schema or definition was agreed upon during a series of workshops attended by leading architects for numerous vendors of collaborative technology, section 4.2.3.2.
- That data could be transferred between systems without fundamentally changing the integrity of the data. Exporter and Importer checklists were created which allowed data leaving one system and data imported into another to be validated, Appendix F. During the implementation stage the proposed standard was tuned at a series of workshops, sections 4.2.3.3, 4.2.3.4 and 4.2.3.5, utilising feedback from participating vendors.
- That data could be stored in a format outside of the collaborative systems which was accessible to collaborative systems deployed on different platforms and built using different technologies. XML was selected to be the vendor neutral storage format as it has become the leading data exchange format.

#### **4.2.3 INDUSTRIAL WORKSHOPS AND SOLUTION DESIGN**

The structure of the research was focused around workshops which were attended by the lead architects of participating vendor organisations, during these sessions any work identified would be allocated and carried out before the next meeting. The following paragraphs describe the highlights of each workshop and then detail the actions carried out between workshops.

#### **4.2.3.1 Feasibility Workshop 17<sup>th</sup> July 2003**

The primary aim of this workshop was to examine the feasibility of migrating data between different collaborative systems, and if possible to produce a rough estimate of the effort required to both design and implementation of the solution. This estimate was required by the participating vendor organisations to ensure that enough time would be allocated for successful implementation. The workshop was attended by seven technical architects, representing all six of the different NCCTP member vendors, the technical lead from the NCCTP steering committee and the research engineer.

Through a broad discussion of how data was stored within each system it was decided that the differences which existed could be overcome and a single generic description of a project stored on a collaborative system could be made. With the feasibility established the group outlined the key activities required to design and then implement a data exchange standard which would allow projects to be migrated between heterogeneous systems.

Prior to the next workshop session, existing work in the field of data exchange involving collaborative tools used by the construction sector was examined, in the hope that it could be utilised as a basis for the NCCTP activities. The 2 papers, addressing the potential impact on vendors and user, Appendix A and B, outline the existing methods available for forming the foundation of transferring data. Some similar activities were found however, under detailed evaluation they were deemed not suitable for the NCCTP requirements without the need to change core functionality of the participating vendor's system.

#### **4.2.3.2 Design Workshops September 2003 – July 2004**

With the review of the existing available solutions showing that they did not meet the NCCTP's requirements for bulk project data exchange a series of design workshops were undertaken in 2003 to create a generic model of a collaborative system which each vendor was capable of storing their data. After each workshop, the findings were constructed into XML Schema to be discussed at the next meeting. By the end of 2003 the NCCTP had a working first version of a bulk data exchange standard, with each vendor investigating the implementation requirements and how well they could support it.

With a new vendor 'Business Collaborator' joining the NCCTP in January 2004 a fresh look was taken at the first version of the standard. The contribution of a fresh member who had not been involved in the initial work proved useful, along with the experiences of vendors that looked to implement the standard, in suggesting improvements which should be made. Feedback from each vendor collected at the session was incorporated into the standard after the meeting to be reviewed at the next design workshop.

Participants at the final design workshop suggested some minor amendments, listed below, to elements contained within the schema which were made then circulated for acceptance.

- New optional description element added to the folder definition.
- Allow for separate XML files which define a document to be validated with the same schema.

- New name element, and extensibility added to the definition of an alias/shortcut.
- Adjustments made to the naming convention used for describing object audit history.
- Allow optional audit history to be defined for a folder object.

Once each vendor had notified that they were happy with the schema implementation efforts were started, with progress to be reviewed at the first implementation workshop.

During the implementation efforts, the standard was adjusted slightly as each vendor mapped their system to the common model, resulting in the officially ratified version of the NCCTP bulk project data exchange standard, Appendix E. The XML schema created by the NCCTP defines the generic collaborative system model as a series of object classes and interrelationships which have been presented in a number of papers, Appendix A – Key Classes and Class Interrelationships, Appendix B – The NCCTP Standard and Appendix D – Generic Collaborative System Model.

#### **4.2.3.3 Implementation Workshop 19<sup>th</sup> October 2004**

The primary aim of this technical workshop was to review the continuing implementation activities of the participating vendor organisations and to address any issues that had been discovered during this. It was established that 6 of the 7 participating vendor organisations were capable of exporting data from their systems to the NCCTP Data Exchange Standard, and had produced sample export data for other vendors to test with. These exports had been imported into 2 of the 7 participating vendors systems, with 4 more known to be working towards support for data importation.

During the meeting it was agreed that the schema would be extended to include a series of true/false elements which would describe the content of the exported data; for example an element(s) to specify the objects for which user rights were stored, or the object level where audit information was held. Participants believed that this would help the importing vendor to quickly identify the most appropriate methodology for importing the data into their system.

During the meeting the technical representatives from the NCCTP partners agreed upon a new change control process which would be used to manage adjustments that were requested by vendor organisation. Table 4 shows the flow of actions and the person/people who were assigned responsibility for them.

**Table 4 - Schema Change Control Process**

	<b>Action</b>	<b>Responsibility</b>
1	Change requests would be sent to Scott Moses via the Notice Board (Online Discussion Group).	<All>
2	Scott will review each request and add any comments and likely Schema changes to address the request before circulating to all NCCTP Technical Standards Group members.	<Scott Moses>
3	A 2 week "Review Period" would be allowed	<All>
4	All comments to be returned to Scott within the 2 week review time.	<All>
5	Where a consensus exists, the changes would be planned for the next release of the Schema.	<Scott Moses>
6	Where no consensus exists, the changes would be raised at the next Technical Standards Group meeting.	<Scott Moses>
7	Where no consensus exists following the next Technical Standards Group meeting, the changes would be raised to the Steering Group Meeting, who would make the final decision.	<Scott Moses>

To give the data exchange standard added credibility the technical group decided that each vendor organisation must supply documentary evidence that they were capable of

both exporting data to and importing data from the agreed standard. It was agreed that a standard checklist of actions should be produced for both the import and export which would then be completed and shared with the other vendors.

The following actions identified during the implementation workshop were conducted prior to the next scheduled workshop:

- The proposed changes to the current version of the standard agreed at the workshop were submitted to the group for evaluation using the new change control process, consensus was reached and the changes committed to the new version of the standard. The proposed release date of this new version standard would be discussed at the next workshop.
- The import checklist proposal document version 1.4, which can be found in appendix F, was circulated to all participating organisations for approval.
- The export checklist proposal document version 1.4, in appendix F, was circulated to all vendor organisations for approval.
- The structure of the elements to be added to the standard which will describe the contents of the exported data.

#### **4.2.3.4 Implementation Workshop 13<sup>th</sup> March 2005**

The primary focus of this workshop was to review the experiences gained by each organisation in the implementation of 1<sup>st</sup> proposed version of the data exchange standard. At this meeting it was established that another two participating organisations had successfully implemented import functionality to work with their collaborative software.

During the workshop the following changes to the current version of the schema, identified during implementation, were agreed, and would be included in the next release of the schema.

- It would be OPTIONAL to hold “Groups” within “Groups”.
- It would be OPTIONAL to have a “Group” without members.
- It would be MANDATORY that users and organizations that had ever existed within a project who then left the project would be marked as “INACTIVE” and not deleted to retain audit links and so they would be included in any export/import.
- It would be OPTIONAL to include a folder description as well as a folder name.
- The agreed list of elements which describe the contents of the exported data.

To increase the credibility of the data exchange standard the technical group agreed that participating vendor organisations should have implementations of the data exchange standard independently verified by a neutral 3<sup>rd</sup> party organisation. The group then collectively produced an outline of what they felt should be included in any independent testing.

- A project already available within the application to be tested would be selected at random and exported.
- The export process and exported XML data would be tested for compliance to the standard.

- A generic project, previously unseen, would be provided by the tester and imported into the application being tested.
- The import process and imported project would be tested for compliance with the standard.

The following actions identified during the implementation workshop were conducted prior to the next scheduled workshop:

- The changes to the data exchanged standard which had been agreed at the workshop were implanted and a new version of the standard submitted to the group for approval.
- An updated version of the import checklist document v1.5, located in appendix F, was produced which addressed concerns raised by several vendors about the wording of the tests at the last workshop meeting.
- An Updated version of the export checklist document v1.5, shown in appendix F, was produced which adjusted the wording of several of the specified tests.

#### **4.2.3.5 Implementation Workshop 27<sup>th</sup> July 2005**

The primary focus of this technical workshop was to finalise the last revision of version one of the bulk data exchange standard which would be utilised as the basis for the independent testing of vendor implementations. At the workshop it was established that every participating vendor now supported the current version of the standard with import and export functionality for their product.

The group then created a series of recommendations which would be required for the independent testing of a vendor's implementation of export functionality compatible with the NCCTP bulk data exchange standard raising the following points:

- Each organisation must have a project ready to export from their system that includes all elements that they support, elements to be compared to completed 'classes and interrelationships' document returned to the NCCTP technical group.
- Scalability testing of export utility on a project that must contain at least 10000 documents.
- Export checks to be carried out on only the sample project, as this will include all functionality supported by the organisations system.

Then, the following recommendations for the tests which should be conducted on a vendor's NCCTP bulk data exchange import functionality.

- Scalability testing of the import utility on sample project that contains a minimum of 10000 documents.
- Independent tester to select one of the sample projects exported from another NCCTP members system to be imported.
- Checks to be carried out only on the sample project not the large project.
- The imported project to be checked using the import checklist document along with project statistics and/or screenshots provided by the source system.

The following actions identified during the implementation workshop were conducted prior to the next scheduled workshop, with a proposal on how independent testing should be conducted circulated to all NCCTP member organisations, see Appendix E.

- Creation of a very large sample NCCTP project to be used for vendor scalability testing and eventually independent testing.
- Proposal on what activities should be included in independent testing, the order of these activities and the likely duration of this testing.
- Creating new versions of example projects which validated against the ratified version of the NCCTP bulk data exchange standard.

#### **4.2.4 IMPLEMENTATION OF BULK DATA EXCHANGE STANDARD WITH CAUSEWAY'S COLLABORATIVE SOLUTION**

As part of Causeway Technologies support for the NCCTP in general and the Bulk Project Data Exchange Standard in particular, the various incarnations of the standard which have been created through its development process have been implemented against the Causeway Collaborative solution. This ability to test the data exchange standard against a real world product at key stages during its development coupled with detailed knowledge of how collaborative systems are actually utilised has lead to a generic standard applicable to the construction sector.

As Causeway's collaborative solution is built upon Livelink the world most used Enterprise Content Management software a number of generic import export utilities already existed for Causeway's Collaborative product. The initial implementation of the standard was attempted using XSLT Transformations which turn NCCTP XML into XML capable of import into the Causeway system using Livelink's standard XML

import, and vice versa. However this approach was unable to deal with the self contained structure, content and users, of the NCCTP defined project, as well as a few other features of the NCCTP schema.

As the analysis of all existing techniques had proved that they were incompatible with the NCTTP schema it was decided that two separate applications would need to be written to control the import and export to and from the NCCTP Standard. These utilities utilised the Livelink API to add to or retrieve information from Causeway's collaborative environment.

When inter organisational testing on the standard started it was found that the flexibility needed in order to allow for each system to map successfully made it difficult for a single implementation of the import application to be flexible enough. Therefore the NCCTP component was separated from the import application allowing greater flexibility in how data was handled during import into the Causeway system.

In addition to providing the framework of a construction project held within a collaborative system and the definition of the key components, the schema provides the ability to extend the data exported from any system. To test this extensibility a project was undertaken which attempted to export all information stored within a collaborative project to the NCCTP Standard utilising the extensibility provided. Although largely successful it was not possible to include every piece of data in a compliant export, these areas have been documented and could form the basis for extending the core Schema already defined.

The work done during the implementation efforts and the live project migrations highlighted in section 4.2.6 have generated a large volume of knowledge around the practicalities of transferring project data between different collaborative systems which is presented in Appendix C.

#### **4.2.5 PRACTICAL USAGE**

With the much anticipated consolidation of the collaboration systems marketplace foreseen in the early 2000's having failed to materialise within the UK, the practical application of the NCCTP transfer has not been tested in a real world environment between two separate vendors. With the designed schema and transfer mechanisms acting as a safety net which prospective clients can rely upon worst case situations.

Since the invent of the data exchange solution by the NCCTP members and it's public launch to members of the construction industry at the Project Extranets V Conference many new ITT (Invitation To Tender) documents for collaborative systems have included cross vendor data transfer requirements. Clients are looking not only for the safety net that the data exchange standard offers but data transfers between collaborative systems at key stages during the project process. For example if a client used an enterprise collaborative system internally for their organisations, but used a variety of project extranets on individual projects they could have the requirement to transfer their latest policy documents to each project to share with other participating organisation. Further examples of the potential uses of the bulk data exchange standard are illustrated in the standard document which is attached in Appendix E of this thesis.

One of the most important uses of the bulk data exchange standard was that the work required to define the key commonalities which existed between the different vendor systems, produced a foundation which other extensions could be built upon. Incremental data exchange between different systems used many of the classes and metadata defined in the bulk data exchange to describe objects and information which had been updated. Real time integration would extend this concept further to make the amended data available across systems as the modifications occurred.

As shown in methodology section 3.1.2, the bulk data exchange standard work is the basis on which all other work conducted during the project is based.

#### **4.2.6 CASE STUDIES**

The following two case studies show the application of the bulk data exchange standard into real world situations.

##### **4.2.6.1 Transferring Data from ASP to Enterprise System**

Since 2004 HBG, one of the UK's leading construction services organisations, have run an internal collaboration system based upon 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, details below, 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 was being transferred between instances of the same system.

Project Parameter	Value
Number of Users	115
Number of Folders	913
Number of Documents	2231
Number of Revisions	3703
Physical Size of Revisions	667MB

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 for transfer 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, 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.

#### 4.2.6.2 Transferring Pilot Projects to Live Environment

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. The table below details the metrics of the projects migrated from the pilot server.

Project Parameter	#1	#2	#3
Number of Users	86	86	86
Number of Folders	1506	435	432
Number of Emails	912	919	294
Number of Documents	6634	2088	1234
Number of Revisions	7213	2259	1365
Total Size of Revisions	2.32GB	305MB	249MB

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 the table below.

- 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.

Activity	#1	#2	#3
Export Project Data	75min	30min	15min
Validate Exported Data	20min	20min	20min
Transfer Data	15min	7min	5min
User Account Mapping	60min	(same users)	(same users)
Import Project Data	135min	60min	35min
Validate Imported Data	20min	20min	20min

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.

#### 4.2.6.3 Case Study Findings

The two practical applications of the bulk project migration showed that it was of primary importance to get the user and group mapping information mostly completed prior to starting the migration, as creating and mapping users between two different systems cannot be automated and that it takes a long time. It was also found that importation or project data was vastly more intense than exportation, and took a greater length of time to complete. The conclusions from these case studies and their implementations for the transferring of bulk project data is discussed at length in appendix C.

## **4.3 ARCHIVING IMPLEMENTATION OF BULK DATA EXCHANGE**

Archiving is the extraction of project data from a collaborative system upon completion of a project which provides an encapsulated collection of all content shared on the collaborative system during the project. Giving access to all the data without the need to continue to pay for the project to be hosted on a collaborative system is an ideal solution for minimising licensing costs in some payment models. However this solution is not without problems with each vendor supplying a different type of archive, with a unique GUI and access method. The collaboration of vendors on the NCCTP Bulk Data Exchange Standard offered the ability to create an archive solution that would be vendor neutral, since data could be extracted from projects in NCCTP then adjusted to the structure required for archive viewing. This sub project firstly examined the feasibility of this solution then investigated how it could be implemented.

### **4.3.1 PROJECT ARCHIVING NEED**

As discussed earlier the ability for all project participants to continue to have access to project related data after the completion of the work is very important, however the costs of enabling this read only access to the data needs to be considered. To meet this need the vendors have created a number of different solutions which give the ability to view the data which was stored on the collaborative system. However this information is normally only given to the party which entered into the contract with the vendor therefore other parties are unlikely to get any access to this data after the completion of the project. Therefore there is a current requirement to find an inexpensive mechanism

which will allow for all participants on the project to have the ability to access all content they had rights to view after the completion of the project.

As the number of different collaborative systems with which companies have to work is showing no sign of reducing, archived data from projects will remain inaccessible to the company employees as a whole due to the vendor specific way in which the data is presented. This in turn will lead to silos of knowledge within the company and reduce collaboration across the enterprise. So there is a requirement to get the data in a format that is configurable so that archives from different vendors can be presented to employees in a uniform way.

#### **4.3.2 VIABILITY ASSESSMENT OF SOLUTIONS**

Any generic archiving solution which was developed would need to have the flexibility of presenting the same data in a number of different ways, so as to provide the different project participants with a customisable viewer for the data. The NCCTP Bulk Data Exchange Standard would form an ideal foundation for such an archiving solution as the structure and contents of the data are very similar independent of the system which has been used to generate data. When accessing data from the archive the speed in which the correct data could be found would be an important issue to consider.

Potential problems would be encountered if a viewer was placed directly onto the data extracted via the bulk data exchange mechanism due to the following reasons:

- The NCCTP Schema used to validate the XML of exported projects allows for the definition of document objects to be included in separate XML documents,

with the relationship between parent and child held within the folder container. Any archiving viewer would therefore need to open a series of XML documents in order to render a folder view with the performance overhead which that would entail. This is especially important when you consider that the data could be stored on slow optical media such as CD or DVD.

- For large project structures, huge quantities of data would either need to be kept in memory or processed each time a new view of the information was requested. This is due to the entire structure of the project being included in a single XML file, as this is the best mechanism for holding bulk project data, when access is required to the data in its entirety.

To solve this problem two different approaches are possible, the first would be to change the structure of the XML produced by the bulk extract, and the second would be to create the archiving structure directly using only the definitions of classes from the bulk exchange standard.

#### **4.3.3 UTILISING BULK EXCHANGE STANDARD FOR ARCHIVING**

Using the elements and structure defined by the NCCTP Standard as a foundation for an archiving solution is logical since it contained definitions of the objects contained within the collaborative system, meaning that every vendor could understand them. The actual output from the bulk export could not be used, without manipulation, as the source data for the archive views due to performance issues which would be faced on large projects phasing the entire xml structure.

Project data extracted to conform to the bulk exchange standard contains all the information which is required from an offline archive including:

- Browse the project folder structure
- Searching for objects via metadata
- Access to document revisions
- Access to audit Information
- And access to the physical files, for each revision.

The advantage of separating the data from the presentation layer means that generic style sheets can be written to display information which had been extracted in NCCTP format from any collaborative system, thus data from any system would appear in the same format.

#### **4.3.4 IMPLEMENTATION**

The implementation of the archiving solution was split into a number of distinct work packages that would deliver a working archive prototype which could be applied to NCCTP data. Once the selected solution to adjust the structure of XML required for the archive was taken, it was required to adjust the extraction to produce the new structure, create presentational views of the data, and then test it, with the implementation of each discussed in more detail below.

Prototyping the archiving solution would be done using Causeway's Collaborative product and the Data Export Capabilities of the NCCTP Bulk Data Extract Tool.

#### **4.3.4.1 Archive Format Data Extraction / Transformation**

To create the optimum performing archive structure changes were required to the XML produced by the NCCTP Bulk Extraction Utility. To investigate the work required for vendors to implement the solution it was necessary to implement both the possible solutions.

- Firstly Causeway's existing NCCTP Extraction utility was adjusted to produce the XML structure, which represents a vendor independent archiving solution, i.e. the archiving extractor could not be used by another vendor even if they were a member of the NCCTP.
- Secondly a Transformation utility was written that would take an NCCTP Bulk Data Exchange Source XML Document and transform it to the format required for the archive. This solution would potentially be a vendor independent solution as it could be used by any vendor who could produce NCCTP Bulk Data Exchange compliant data.

The implementation of the adjustments to the extraction utility proved far easier than the transformation of the NCCTP Bulk Exchange source, as the content of the source could change substantially depending upon which vendors system had generated it. This was later helped by the addition of additional XML Tags to the source document that identified the characteristics of the system which generated it.

#### **4.3.5 PRACTICAL USAGE**

Project Collaboration systems host projects of all sizes, which once completed are effectively removed from the online collaborative tool, with the exception of those

which are continued to be paid for. At completion a project's data maybe archived by some vendors and sent to the client they have been contracted by, while those who simply participated on the project will receive no data from the system.

An XML/XSLT Based solution which displayed data inside a browser would offer a low cost method of delivering project data to all participating organisations. Since any set of style sheets could be applied to the data a common look and feel could be obtained across projects which were originally hosted on different collaborative products.

#### **4.4 INCREMENTAL PROJECT DATA EXCHANGE**

The work done in this area of the project was an investigation into the feasibility of enabling project data transfer between different collaborative products, at regular intervals throughout a project. Through the execution of this sub project the requirements of both client's and solution providers were considered to ensure that the proposals arrived at would deliver what clients required while minimising the efforts of the providers. On completion of the investigation the proposals were constructed into a document for incremental project updating which was submitted to the NCCTP Technical Committee for consideration as a standardised way to transfer data between collaborative systems.

##### **4.4.1 THE NEED FOR INCREMENTAL DATA EXCHANGE**

From the perspective of the vendor's participating in the NCCTP initiative incremental data exchange provided a steppingstone on the route to real-time data exchange, the long term goal of the group. Incremental Data Exchange was seen as a valuable step as

it would examine some of the same issues involved in real-time integration, and add to the knowledge of their organisations.

Incremental data exchange solution would enhance the usefulness of collaborative products to clients by.

- Allowing project data to be held in a number of different collaborative systems simultaneously, so individual companies could view the data in an environment they were familiar with.
- The ability to verify a bulk exchange between systems without the need to turn off one system, because data could be added incrementally.
- The ability to update a number of different projects with the same information stored on a company's own internal collaborative product.

#### **4.4.2 VIABILITY ASSESSMENT OF SOLUTIONS**

Due to the resource constraints that face collaborative software providers along with any other business any solution proposed to, and ultimately endorsed by the NCCTP would need to build on the work already done in the bulk data exchange standard to have a realistic change of being implemented. Therefore a successful solution would have to seek to meet the following criteria, which were agreed at the incremental project inception workshop, section 4.4.3.1.

- Use the same agreed definitions, object classes and properties, as the bulk data exchange standard.
- That the structure in which data was stored should be as close as possible to that of the bulk data exchange standard.

- That the export utilities written for the bulk data exchange extraction could be modified to incremental extraction with the minimum of effort.
- That incremental data could be imported into a collaborative system utilising the same import utility as the bulk data exchange standard, with minimal changes required.
- That the updates generated by the exporting system contained the minimum amount of data required to synchronise both systems.

Due to the conflicting nature of the above points it was not possible for a solution to fulfil all completely, and it was the correct balancing of the various requirements which would lead to the best solution for the NCCTP.

#### **4.4.3 INDUSTRIAL WORKSHOPS AND SOLUTION DESIGN**

The structure of the research in this sub project can generally be partitioned into three distinct sections. The first part primarily focused on examining the feasibility of the work and outlining the criteria required for a successful system. This portion was done in technical workshops similar to the workshops undertaken for the bulk data exchange standard. The second portion of the work focused around the design, testing and documentation of the proposal which was undertaken within the sponsoring company. Finally the proposal created was an extension of the bulk data exchange standard which would allow for incremental data to be exchanged between systems. This proposal was presented to the NCCTP technical group for evaluation at a workshop event.

##### **4.4.3.1 Incremental Updating Project Feasibility/Outline Workshops**

A portion of each of two NCCTP technical workshops was used to examine the feasibility of, and then plan the requirements of the incremental project data updating

solution. The participants of these workshops included technical representatives from NCCTP member organisation who had specialised knowledge of both how their products functioned and how they were used by their client's.

The ultimate outcome of the session around the feasibility of incremental updates was to agree that it would be possible for them to be successfully exchanged between systems.

This decision was reached by the group going through the following process.

- Firstly a common understanding of the meaning of incremental project updating was decided between all participants.
- A high level discussion of the data which each vendor could extract from their system that could be of use in an incremental updating transfer of data between systems.
- A high level discussion around that data which would be required by each vendor if they were required to incrementally update a project which already existed on their system.
- Review of data transfer requirements between the systems, engaged in the incremental updating scenario.
- Decision of the feasibility of the proposal.

The time at the second workshop was used to discuss what was required from the proposal, which is outlined in section 4.4.2, and a brainstorming session and discussion on incremental data updating. The session outlined several points or questions which needed to be considered when designing the proposed solution, which included:

- How synchronisation conflicts, which could exist if data is updated on both systems between incremental updates, would be handled?

- How a record would be kept of objects created by prior incremental updates?
- How the first update would be made between the collaborating systems?
- Potential uses of incremental updating?

#### **4.4.3.2 Incremental Updating Standard Solution Design**

The proposed solution for incremental interoperability allows for data to be synchronised between two different project collaboration tools, however only one of these tools can be active, the second must be read only. This is a requirement because of the issue of potential data conflicts between the two systems, which could occur if data is amended in both systems between the incremental updates. For example a new revision is added to the same drawing in both systems between incremental updates, creating a potential conflict.

Each project collaboration tool has its own unique way of storing object data in its system, which usually takes the form of an ID value. When data is being incrementally transferred between systems a record of object mapping data between the two systems must be maintained. This is because each incremental update will reference objects which have been previously passed in other incremental updates. Further user, group and organisation mapping information must also be maintained, so that the links between objects and users are the same when data is imported into the destination system.

#### **4.4.3.3 Proposal Presentation Workshop**

At the completion of the design for incremental data exchange the created proposal was presented to the other members of the NCCTP for consideration. This presentation of

the standard was done at a technical workshop, which then allowed any questions to be answered and a discussion of the proposal to be held. During the workshop the solution presented was evaluated against the criteria that were set out at the project inception, section 4.4.3.1.

The feedback given towards the proposal by the participating vendors was generally positive; with the feeling that the proposal could form the basis of incremental data exchange between collaborative systems should it be requested by clients on a project. It was noted that the group felt further evaluation of the proposal would be required once implementation efforts were started, similar to that undertaken during the implementation of the bulk data exchange standard.

#### **4.4.4 DEMONSTRATION PROJECT**

To enable the proposal created for incremental data exchange to be tested for compatibility with other NCCTP system without full implementation in these systems, it was necessary to construct a collaborative system which was based solely on the elements described in the bulk data exchange standard. Once in place, the effect of particular incremental updates could be examined, and adjusted if required. The two systems constructed were simple database schemas containing the NCCTP defined elements, with one system populated initially, and the other system by a bulk update.

#### **4.4.5 SUBMISSION TO NCCTP GROUP**

Work on incremental exchange done was collated into a proposal document, appendix H, and presented for consideration to the NCCTP technical committee.

## **4.5 REAL-TIME PROJECT DATA EXCHANGE**

The real-time access and exchange of data between collaborative project solutions was examined in two distinct projects during the EngD. The first looked at providing generic access to manipulate data stored in project collaboration system through a generic API, done as an MSc Dissertation Project (October 2003 – September 2004). While the second built upon the first and other work in exchanging project data to investigate how multiple project collaboration tools could share data on the same project (December 2005 – November 2006). The following two sections examine the work done and key findings of both these projects and show how they relate to other projects undertaken during the entire EngD.

### **4.5.1 WEB SERVICES BASED API FOR COLLABORATIVE SYSTEMS**

In the early 2000's web services were emerging as a new existing framework that allowed integration between applications which were independent of the technologies they were built in or the environments they were deployed upon. Naturally then a web services foundation was selected as the basis for the Collaborative System API designed during the MSC Project. This designed API gave other software applications access to the core functionality contained within collaborative systems in a generic manner which would facilitate one-to-many integration. The following sections highlight the need for the generic API, the viability of the generic API solution, how the NCCTP bulk data exchange standard was used during the project, the implementation of the API with the Causeway Collaboration product, and a demonstration project of another system interacting with a collaborative system via the API.

#### 4.5.1.1 Generic API Access Needs

Since most people's experience of project collaboration tools is as a service offered via the internet, the need for generic API access to the data, or the ability to interact with the project workspace through an API is hard to see. Indeed even the tools which were deployed globally for a particular client or construction organisation provided their own API's which allowed application integration between the collaborative software and other systems. One vendor, 4Projects even published a web services based API for programmatic access to their collaborative system (4Projects Desktop, 2005), although access has since been restricted to members only.

This approach to providing integration with other application does however have a number of problems for both 3<sup>rd</sup> party software suppliers and clients of collaborative software:

- Not every vendor has an API, meaning that it would not always be possible for clients to expose their functionality to other software applications utilised in their business. Indeed as many clients work on multiple different tools concurrently, integration would not be feasible since it could not be easily accomplished across the board.
- The vendor API's which existed utilised very different technologies, effectively meaning that any integration would have to be done on a one-to-one basis.
- That the APIs offered by the vendors of collaborative software each offered their own unique set of functionality, this inconsistency between functionality meant that integration would be more difficult.

- That collaborative system API's were not always available publicly, meaning that 3<sup>rd</sup> party software vendors could not build integrations which would work with a number of different collaborative products.
- That APIs offered by individual vendors could evolve overtime meaning that integrated software would have to be updated to work with the newly created API specification for that vendor.

The Generic API for interacting with the collaborative project workspace would overcome these identified problems through

- The system integrator would have access to an API which worked against a number of different vendor's software, and that as long as any integration was done against the latest version of the API then this would function against newer versions of individual vendor's products.
- A single integration solution could be designed and built to work with collaborative products, since the technology used for the integration is not tied to a particular technology.
- That a single set of functionality would be available against each vendor's solution, as a generic solution had been implemented.
- System integrators would not need access to every different product to create generic integrations, saving costs and producing a better solution.

#### **4.5.1.2 Viability Assessment of Solutions**

A major part of the early work done in the project was to investigate the feasibility of designing a generic solution which would allow applications to integrate with collaborative software. As many individual vendors had already exposed their

collaborative solutions functionality to 3<sup>rd</sup> party integration then the core functions needed to be included were easier to identify and to include.

- The ability to add and retrieve document revisions from the collaborative system.
- The ability to view and update the metadata associated with objects contained with the collaborative system.
- The ability to search the collaborative system for objects and within document revision for content.
- The ability to browse the structure in which objects were stored in the collaborative system.
- The ability to authenticate against the collaborative system.

The bulk data exchange standard which defined the generic structure of collaborative systems was used as the foundation of the different API requests as it defined each of the objects required in the transactions. Its Document / Revision / File structure allowed conceptually for the addition and retrieval of document revisions to and from a collaborative system. The class definitions for each object within the bulk data exchange standard allowed for the set of metadata viewable and updateable in the collaborative system to be defined for each object. Search queries and result returned from these queries could be standardised using the definitions contained within the bulk data exchange standard as they were understood by different vendors. The structure contained within the collaborative system project could be browsed conceptually since the bulk exchange standard defined the structure. Additionally, user authentication would be possible since the classes for Organisation, User and Group were already defined.

### **4.5.1.3 Extending Bulk Exchange Standard to API**

The Generic API for accessing functionality in collaborative systems from other applications comprises a set of request and response transactions. The elements contained in these transactions are taken from those defined in the NCCTP Bulk data exchange standard, which allows for a common understanding by different vendors of collaborative software. Since 3<sup>rd</sup> party applications do not store data which is similar to that stored by collaborative solutions it is not required to maintain any object mapping information. However user authentication is required in order to verify that the user can access or modify the content. This is not covered in the Bulk exchange standard and hence this has been designed into this solution.

The following list outlines the API's which have been created:

- Add Object to Collaborative System
- Add Document to Collaborative System
- Add Document Revision to Collaborative System
- Download Object from the Collaborative System
- Download Document Revision from the Collaborative System
- Get Allowable Search Criteria from Collaborative System
- Get Allowable Object Metadata form Collaborative System
- Get the Children of an Object in the Collaborative System
- Get the Metadata associated with an Object
- Search the Collaborative System
- View a Document Revision

#### **4.5.1.4 Implementation**

For the implementation of the generic APIs a Java web services layer, deployed in Apache Axis, was written for Causeway's collaborative product which exposed the stored project data to other applications. This web services layer utilised an existing repository API provided as standard by the application. The web services layer written for this project to create the generic API, controlled reading and responding to HTTP request sent over the web.

#### **4.5.1.5 Demonstration Project**

The demonstration project designed to simulate a 3<sup>rd</sup> party application interacting with a collaborative system through the designed generic API was a client based text editor. This simple text editor application which was written in java would demonstrate the ability fetch, add revisions to a collaborative system and browse the structure in which the content was stored. These functions would replicate common desktop actions when using applications like word, excel, autoCAD etc. when a user open a document / drawing revision, amends it and then saves the newly amended file. The collaborative system used as the document repository was Causeway's collaborative product, which had been extended to include the generic API, section 4.5.1.3.

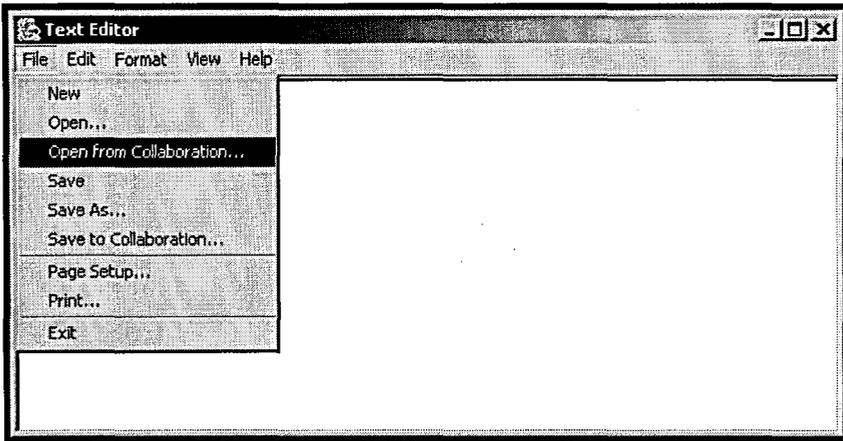


Figure 7 - Opening a Text Document from Collaborative System using API's

In the demonstration project the user would start the text editor application and then select to open a document from the collaborative system, Figure 7. Once selected this document would be fetched to the user machine via the document revision fetch generic API function, and opened in the text editor. The user then edited the content of the document, and selected to save it back to the collaborative repository, Figure 8.

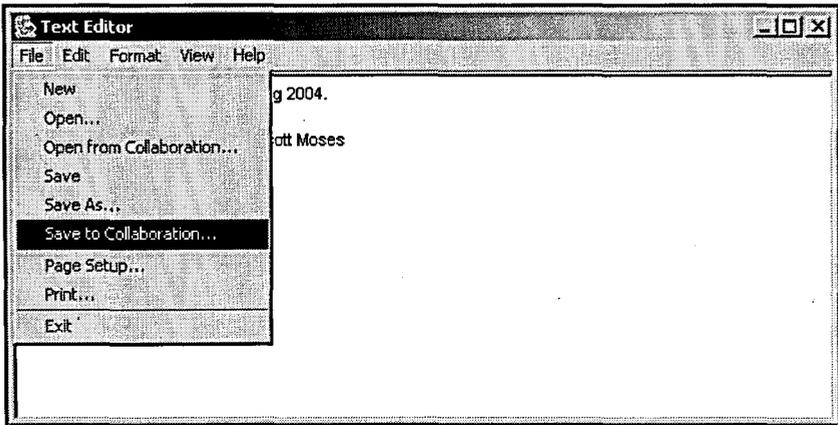


Figure 8 - Saving Document Back to Repository using API's

The Application then allows the user, via the generic API browse functionality, to browse the collaborative repository to locate where they wish to save the new

document, Figure 9. Once located the user submits to file to the collaborative repository, using create revision generic API function, and it is saved in the correct location in the repository. The uploaded file can now be accessed by other users of the collaborative system through the applications web interface, Figure 10.

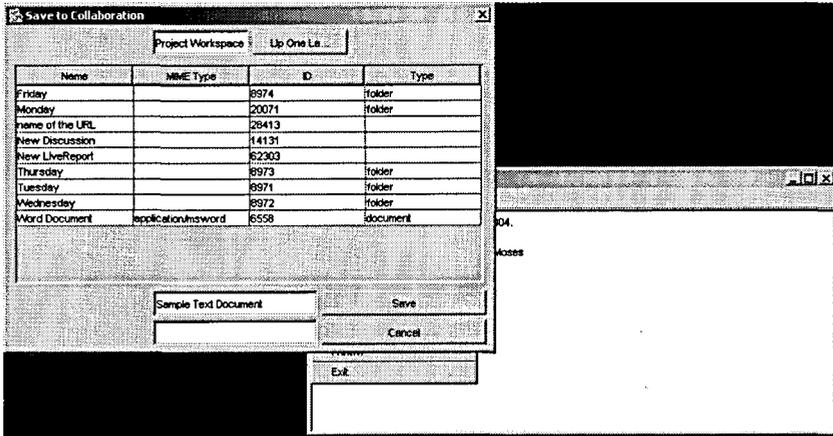


Figure 9 - Browsing Collaborative Repository using API's

Through the scenarios tested the demonstration project successfully showed how the designed generic API could be used to allow integration between 3<sup>rd</sup> Party Applications and collaborative systems.

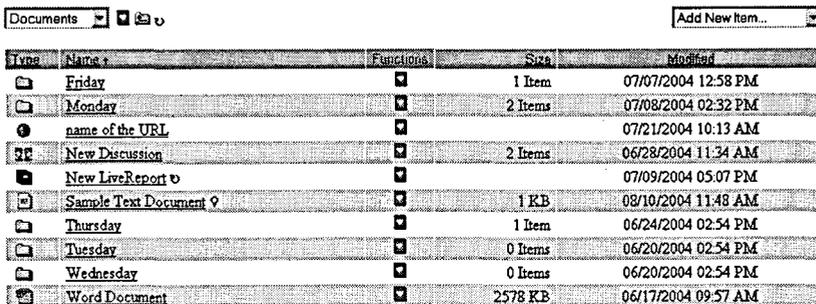


Figure 10 - Interacting with Document Created via API's in Standard Web Interface

## **4.5.2 REAL-TIME EXCHANGE FRAMEWORK**

Previous components within the overall EngD project had focused on the transfer and or duplication of the project data stored within a collaborative system, the real-time exchange framework investigated within this sub project aimed to give access to data without the need to duplicate information. The real-time exchange framework was identified by the NCCTP technical committee as one of the future projects of interest which could bring real benefit to client's of collaborative solutions. In the following sections the current needs for a real-time exchange solution are presented, along with a viability assessment providing real-time integration capabilities. It shows how the solution design was made, and its proof of concept implementation and demonstration project. Finally it details the proposal handed over to the NCCTP technical committee for consideration, and the next steps along the road.

### **4.5.2.1 Clients Real-Time Needs**

Extending the need set out in section 4.5.1.1 for real time access to a collaborative repository for 3<sup>rd</sup> party applications, access to collaborative data from other collaborative systems will bring the following benefits.

- The user will be accessing data in a familiar user interface.
- No additional training requirements for users when a new project starts and a different solution is selected, for storing the project data.
- Organisations which have invested in enterprise system for managing their internal and external work will have the ability to manage everything on a single system.

#### **4.5.2.2 Viability Assessment of Solutions**

Any solution that aimed to give users access to information stored in a number of different collaborative repositories would have to overcome issues relating to the fact that content is dispersed across multiple systems. The following list shows the main points which would need to be addressed by any proposal:

- Since collaborative systems are based around the principle of controlled access to the contents stored within their repositories, any integration framework would require users to be accurately identified and authenticated. Either through each system holding a common set of user credentials, or mappings between user accounts on different systems.
- The mechanism selected for the integration must have the capability to integrate software which is built utilising different technologies and deployed in different environments.
- Must define transactions using terminology which is understandable to all vendors of collaborative systems.
- Provide fast and reliable access to the information stored in repositories

#### **4.5.2.3 Industrial Workshops and Solution Design**

After the successful implementation of the bulk project data exchange standard by the participating vendors the format of the technical workshops was adjusted, splitting them into two sections. One continued to examine the bulk project exchange standard, while the second discussed future interoperability projects, such as extending bulk exchange, incremental exchange and real-time data access.

The discussions at the technical workshops relating to real time interoperability, which outline the technical requirements of each vendor, were used in the creation of the proposed standard shown in appendix I.

#### **4.5.2.4 Implementation**

Currently implementation of the proposed real-time data exchange standard has only been done with one collaborative tool, Causeway's collaborative system. This implementation has allowed the design solution to be tested against a real collaborative system used by the construction industry. Through the demonstration project 4.5.2.5 and linking together two Causeway collaborative systems basic real-time interoperability has been proven to work.

Through the knowledge and understanding gained by the author, in the implementation of the NCCTP's bulk project data exchange standard by multiple vendors, it is expected that the proposal will require modifications as vendors attempt to implement this in their collaborative systems. It is also expected that further adjustments to the proposal will be required once intra-organisation testing starts.

#### **4.5.2.5 Demonstration Project**

The demonstration project undertaken was designed to prove that the proposal for real-time integration between heterogeneous collaborative software would allow for the client requirements to be met. This prototype involved the creation of a collaborative system which was based upon the common definition defined in the NCCTP bulk data exchange standard. It was assumed that if an actual collaborative system could integrate in real-time with this NCCTP based system then other could integrate in real-time using

the same methods. This was because all vendors had already shown they could map to common definition when they implemented the bulk project data exchange standard.

The main advantage of this approach is that it allowed the transactions to be validated, without the need to involve another collaborative tool, which would have delayed the demonstration project. Thus the real-time integration solution tests were carried out between the created NCCTP based collaborative tool and Causeway's collaborative software.

When operational the demonstration project showed that the proposed standard for real-time integration allowed for users in one system to browse and interact with content stored in the other system, without the need to replicate the data in two repositories.

#### **4.5.2.6 Submission to NCCTP**

The work done on the creation of a proposed standard to enable real-time access to content stored in different collaborative repositories, Appendix I, has been shared with the other members of the NCCTP but is yet to be subjected to formal evaluation. This proposed standard was produced to meet one of the stated requirements of the NCCTP from a technical perspective, it is currently available to all members as a starting point should they be involved in a project where the client requires this. A formal evaluation of the proposals, by the technical commit of the NCCTP has been put back to 2008.

## **4.6 SUMMARY**

This chapter has given an overview of the research undertaken for the EngD project into increasing the interoperability of project collaboration systems. Each sub project has

been presented separately so outcomes of each can be more clearly identified, however they are intrinsically linked, Figure 2, with each sub project forming the building blocks for the next.

In the first project a bulk data exchange standard was created through a workshop approach involving the lead technical people from all participating NCCTP Vendor organisations. The discussions undertaken at these workshops were transformed into an XML schema, and associated documentation which represented a generic model of a collaborative system currently used by the UK AEC Industry. Through extensive intra-organisation testing and validation, using import and export checklists this standard which will enable bulk transfers of project data between heterogeneous collaborative systems, has been implemented by the majority of vendors. To aid vendor organisations ensuring their implantations are working correctly, an independent verification procedure has been designed, Appendix G.

Building upon the foundations created through the generic collaborative system model constructed, additional applications of the bulk data transfer functionality were investigated, as another project. The first utilised the extensibility within the generic bulk data exchange standard to construct export and import functionality which would include all of the custom data contained within a collaborative system. This was applied to Causeway's ECM solution and is currently used for migrating project data between different instances of the application to eliminate data lost. The second extension of the bulk transfer functionality was into the area of project data archiving, where XML, XSLT and HTML based solutions were created. This utilised a slightly adjusted, to improve performance, NCCTP export as the basis for data views created using XSLT.

The archiving solution generated during this research is used as the basis for Causeway's project archiving solution.

The project then investigated the area of incremental project data transfer between collaborative systems, which would allow data to be updated at regular intervals. The research approached the problem from the view that ideally vendors would wish to utilise their existing import export functionality if possible. The proposed standard which has been created builds upon the generic model created for the bulk exchange standard and customises this to incremental updates, not the most efficient updates but the most practical. The work presented in the proposed standard has been shared with the NCCTP Technical committee, but has not been formally adopted as the NCCTP formal standard which vendors must conform to. Practical validation of the work on incremental integration has been done using Causeway's ECM product.



## **5 CONCLUSIONS AND IMPLICATIONS**

### **5.1 CONCLUSIONS**

This chapter summarises the finding of the entire EngD, and shows the impact the research has had on the industrial sponsor, the suppliers of collaborative solutions and the wider AEC Sector. It continues to highlight further work which could be undertaken to advance interoperability between vendors, from an academic, vendor and client view point. Finally the chapter makes a critical evaluation of the work done in the EngD and presents the main conclusions.

### **5.2 REALISATION OF AIMS AND OBJECTIVES**

The primary aim of the research presented in this thesis was to increase the level of interoperability between collaborative software solutions currently used by the UK AEC Industry, through the creation of standardised data exchange methodologies. With the creation and implementation of the bulk project data exchange standard by most of the leading suppliers of collaborative solutions, coupled with the proposals for incremental and real time exchange, the goal of increasing interoperability has been successfully accomplished. Clients of collaborative solutions are now able to move project data from one system to another, without the need for any custom work on the part of the vendors.

The bulk data exchange standard written and implemented during NCCTP technical project, has been proven to allow migration of data between heterogeneous systems through inter-organisational testing, and validated utilising the checklists in Appendix F. This standard, which fulfils objective 1 of the EngD, has been ratified by all members

of the NCCTP in 2004 as the NCCTP Bulk Project Data Exchange Standard version 1.102.

After the ratification of the bulk exchange standard by participating vendors, research was undertaken to examine the potential additional applications for bulk project data extraction and the benefits these could bring, objective 2. Two different areas of research were pursued, one examining the potential for using the extensibility provided in the standard to support application specific data, with the other examining archiving. The archiving investigation offered potential benefits to both the users' and providers' of collaborative solutions showing the NCCTP extracted data could form the basis of a project archiving solution. This extension would deliver to vendors a cheap uniform mechanism of supply data to all project participants after completion of the project, and to clients the ability to apply a uniform 'look and feel' to data from many different systems. To prove that this benefit was practically obtainable, work was done implementing the proposed solution using Causeway's Collaborative product. The second investigation into the benefits that an extended version of the data exchange standard could bring, showed that it was possible to migrate information above that defined in the core standard, using the extensibility provided. This solution again was implemented using Causeway's ECM product to prove its practical applicability. This showed that clients could benefit from a greater level of interoperability should vendors extend their extraction and importation routines beyond the core NCCTP elements. The NCCTP based archiving solution constructed as part of the research is now used by Causeway for supplying project data to customers at the completion of their projects. The extended bulk project data exchange created is used by causeway for

migrating live customer projects between different instances of Causeway ECM solution.

Objective 3 of the project was to propose a solution which would allow for incremental data updates to be passed between collaborative systems, replicating the contents from one vendor system into another. This objective has been met through the creation of the proposed NCCTP Data Exchange Standard, appendix H. It builds upon the work done in the creation of the NCCTP bulk project data exchange standard, and generally utilises the same generic system model in the incremental updates which exists in a bulk transfer. Indeed a bulk transfer is proposed in the standard as the initial import before incremental updates can begin. Although this use of the generic structure from the bulk exchange standard does not make for the most efficient incremental updates, it does mean that vendors should be able to utilise their existing import export functionality to process and create these incremental updates with only limited adjustments.

Objective 4 of the project was to propose a solution which would allow real-time integration between different collaborative systems been utilised on the same project. The proposed solution which would allow for the same content to be viewed and manipulated through multiple user interfaces is shown in Appendix I. The Web Services based solution outlined has been designed to avoid the necessity to duplicate physical files in multiple systems which was a feature of the incremental exchange solution. This is due to the fact that since most solutions are hosted in data centres there is no time difference from fetching the files from one system as opposed to another.

### **5.3 IMPLICATIONS FOR THE SPONSOR**

Causeway Technologies Collaborative system is fully compliant with the latest version of the NCCTP bulk data exchange standard, allowing projects to be relatively easily transferred either into the system or out to another system. Supporting and fully implementing the bulk data exchange standard will give current and future clients added confidence in the Causeway Collaborative solution and its interoperability with other collaborative solutions offered by other vendors.

Through being at the forefront of the proposals into extensions of the implemented bulk data exchange standard to areas such as incremental and real time integration Causeway Technologies is in a very strong position should these standards be adopted by industry. This is coupled with the proof of concept solutions existing for Causeway's collaborative product which allow incremental integration and real time integration with other NCCTP compliant systems.

#### **5.3.1 BUSINESS EXPLORATION OPPORTUNITIES**

Additional services can now be offered by Causeway Technologies to parties which collaborated on the project, such as the ability to archive their data from the project at completion. Previously only the client would receive this information as it could be an expensive business getting the data out of the system and giving away the ability to view it.

The ability to selectively import data from a number of other collaborative systems into a single repository used for the project will reduce the need to manually transfer data saving time and money for the client, and its partners on the project.

#### **5.4 IMPLICATIONS FOR WIDER INDUSTRY**

The work presented in this thesis is the first attempt to define a series of standardised mechanisms for transferring of project data between AEC focused collaborative systems which had sufficient backing for industry to succeed. By approaching the problem from the perspective of needing to work with existing collaborative systems rather than proposing an idealised model potential barriers to implementation have been removed as core systems do not need to be changed. The work done in this research project to increase interoperability between collaborative systems will increase clients' confidence in the stability of project collaboration tools. Clients will no longer be locked into a single solution for the entirety of the project as data can be migrated to another solution if required.

The proposals for incremental and real-time integration contained in this thesis when implemented by industry offer users the ability to access multiple repositories from a single familiar user interface, increasing system adoption and user productivity.

In section 5 of the 'The NCCTP Data Exchange Standards Impact on the UK Construction Industries Collaborative Technology User Community' conference paper, Appendix B, the impact of the bulk exchange standard to clients is highlighted. This paper argued that although the ability for users to access their data in XML format is nothing new, the bulk exchange standard does allow for project information to be

exported in such a way that another collaborative application can readily interpret the data and recreate the project. Furthermore, clients have the ability to select a system which best matches their needs, as the bulk exchange standard has been implemented by a large number of vendors. The emergence of the bulk exchange standard may have little impact on the system selection decision made by clients, as it has been implemented by the majority of vendors, but it will increase the confidence to use this technology.

The conference paper in Appendix A shows the impact which bulk data exchange will have on the suppliers of collaborative solutions to the AEC sector. It highlights how the standard has reduced the time and effort which would have been required to transfer a project from one system to another, through the creation of a common model which all vendors map to. This eliminates the need to do a one to one mapping of collaborative systems per migration. This common model which all organisations map their own collaborative systems to means that each vendor does not need to be concerned about the changes made in any other vendors system, only the changes made in their own. Since the NCCTP bulk exchange standard does not require any changes to a vendor's core system, it is possible for additional collaborative providers to implement, which was illustrated by Business Collaborator, who joined the NCCTP after the standard had been agreed.

## **5.5 RECOMMENDATIONS AND FURTHER WORK**

The following three sub sections contain recommendations for further work which will advance this project.

### **5.5.1 ACADEMIC RESEARCH**

The NCCTP work on data exchange presented in this thesis focused primarily on delivering a solution to industry which could be easily implemented by the vendors and would not require any changes in their own applications. This resulted in some of the more complex functions in collaborative systems, such as workflow and forms are not covered by the specification. Further research is recommended to devise a method to include these in the existing standard which will not require changes to the different underlying applications.

### **5.5.2 AEC INDUSTRY**

The work presented in this thesis on the bulk exchange of project data, which has been implemented by the collaborative system vendors and subjected to extensive intra-organisational testing, has never been deployed to transfer live data between different collaborative systems. Although the data exchanged standard has been used for the migration of live project data between instances of the same tool, true evaluation can only be achieved once a live project has been transferred between different solutions. Clients of these systems do have to be aware of some of the potential issues of transferring projects between systems, highlighted in the paper 'The Practicalities of Transferring Data between Systems used by the Construction Industry', Appendix C. In its conclusions the paper identifies the following issues which need to be considered:

- That the generic collaborative system model by its very nature will be unable to accommodate all of the custom data and structures which existed in the source system. Furthermore it may not be possible to create these in the destination system due to the differing underlying

structures. Meaning that some loss might be experienced for elements which exist outside of the specified NCCTP schema.

- That the project export is only a snapshot of the project at the time of extraction, and any data added during or after the export may not or will not be included. This effectively means that the users of the collaborative system will only have read only access once the migration has started, increasing the total level of down time for the process.
- That the total duration of the migration cannot be exactly known prior to the completion of the activity, meaning that adequate contingency time must be allotted for the transfer.

The AEC industry needs to continue to push the vendors to increase the level of interoperability between their collaborative products if the incremental and real time data exchange proposals are to be implemented.

### **5.5.3 IT VENDORS**

It is recommended that the vendors, as a group, work towards implementing incremental data exchanges between their products, extending the existing bulk data exchange standard which is already in use. This will allow updates from many repositories into a single collaboration solution which is used as the project collaboration solution. The proposed NCCTP Standard for incremental Project data exchange should form the basis for this effort, although changes are inevitable once implementation begins.

It is further recommended that real time data exchange between collaborative systems, based upon the draft specification contained in this thesis, Appendix I, is implemented by vendors collectively. Should this be implemented then real benefits could be realised by users of these systems as they could access project data which resided in a number of different repositories from a single familiar user interface.

In any future interoperability efforts the importance of the major vendors working together to achieve the goal will be paramount to success.

## **5.6 CRITICAL EVALUATION OF THE RESEARCH**

The main limitations of the research presented in this thesis were that neither of the incremental or real-time interoperability proposals were tested by the different vendors to ensure they were capable of transferring the correct data. This is reinforced by the experiences gained during the implementation of the NCCTP bulk data exchange standard where continual minor adjustments were made based on feedback from participating vendors.

Furthermore, although the bulk data exchange standard, which forms the foundation of this research, has been implemented by the suppliers of collaborative technology to the UK AEC Sector, no live projects have been used to migrate between heterogeneous systems. This has primarily been due to the increasing stability and longevity of the providers and the failure of consolidation of the sector to actually happen. Application of this methodology has been limited to intra-organisation testing between the vendors and live project transfer between different instances of the same application utilising the extended version of the bulk data exchange standard at Causeway.



## 6 REFERENCES

4Projects & Tesco Express 2003, 4Project press release – Tesco Express, 4Projects.com, available: <http://www.4projects.com/Page.aspx?CAT=206>, date accessed 11<sup>th</sup> December 2007

Access To Energy 1998, Disruptive Technology, Access To Energy Enterprise Monthly News Letter, Vol 26, No. 2 – 10/98

Box 2001, Brief History of SOAP, developMentor, <http://www.develop.com/dbox/postsoap.html>, date accessed 11<sup>th</sup> January 2003

Building, 2004, Revenue Probes IT Provider BIW, Building Magazine Issue 37, 24<sup>th</sup> September 2004, p 14.

Butler Group, 2003, Workgroup and Enterprise Collaboration – Reducing the Costs and Increasing the Value of Collaborative Working, Hull: Butler Group

Charlesworth, I., 2002, EAI and Web Services – Cutting the Cost of Enterprise Integration, The Butler Group

Christensen 1997, The Innovator's Dilemma - When New Technologies Cause Great Firms to Fail, Harvard Business School Press

CITSEC, 2004, IT Security in Collaborative Environments, Construction IT Security Forum, <http://www.citsec.com/it-security-in-collaborative-environments.htm>, accessed 5 October 2004.

CMIS 2003, The History of EDI, CMIS Ltd. Knowledge Base, [http://www.cmisltd.com/kb/kb\\_edi3.htm](http://www.cmisltd.com/kb/kb_edi3.htm)

Cohan 2000a, The Dilemma of the Innovator's Dilemma, The Industry Standard, <http://www.thestandard.com/article/display/0,1151,8526,00.html>, date accessed 2002-12-18T11:18:00

Cohan 2000b, The Cases of Schwab and Hewlett-Packard (HWP), The Industry Standard, <http://www.thestandard.com/article/display/0,1151,8526,00.html>, date accessed 2002-12-18T12:12:54

Conry-Murry (2004), WS-Security makes SOAP Safe, Network Magazine, 5<sup>th</sup> June 2004

CoverPages (2004), Updated WS-Trust and WS-SecureConversation Specifications Accompany Microsoft WSE 2.0, 24<sup>th</sup> May 2004, Coverpages Hosted By OASIS

Creswell, J: (2003). *Research Design: Qualitative, Quantitative, and Mixed Method Approaches* (2<sup>nd</sup> edition). Thousand Oaks: Sage Publications, ISBN 0761924426

Cutler, 2006, Information Aggregation, AEC Magazine January/February 2006, pp 20-21,

Dane, F. C. (1990). Research Methods, Brooks/Cole, Pacific Grove.

Dainty, A. R. J. (2004). Methodologies for Construction Management Research Research Innovation & Communication [CVP008], MSc Construction Innovation course notes. Loughborough University, UK.

DataDirect 2003, Web Services in Europe Survey Report – January 2003, Graph G. Obstacles to Implementation?, DataDirect Technologies.

Davis (2003), New SOAP Specs Refine Security

Dyck 2003, New SOAP Specs Redefine Security, eWeek, [http://www.eWeek.com/print\\_article/0,3668,a=36329,00.asp](http://www.eWeek.com/print_article/0,3668,a=36329,00.asp), 27/01/2003

ECOSPACE, 2006, Welcome to the Integrated Project ECOSPACE, [ecospace-ip.org](http://www.ecospace-ip.org), available: <http://www.ecospace-ip.org/>, date accessed: 10<sup>th</sup> December 2007

Egan, J, 1998, Rethinking construction: The report of the construction task force, Department of the Environment, Transport and the Regions, London. p16.

Gartner 2002, Why Google's SOAP Interface Matters, Research Note – SPA-17-2188

Gates 2000, Remarks by Bill Gates, Professional Developers Conference, Orlando, Florida, July 12, online url <http://www.microsoft.com/billgates/speeches/2000/07-12pdc.asp>

Hayward 2002, MapPoint .NET – A Natural Web Service, Gartner Research Note – CS-17-1057

Holden, G. (2001) Outsmarting the Outsourcers, in: Forbes, Special Volume: Best of the Web B2B, 2001, p. 52-53.

Integral 2002a, The Rise and Fall of Disk Drive Manufacturers, Integral-Inc, [http://www.disruptivetech.com/dt\\_examples/dte-drives.html](http://www.disruptivetech.com/dt_examples/dte-drives.html), date accessed 2002-12-10T11:14:00

Integral 2002b, The Story of Steel : Integrated Steel Mills' fall from Grace, Integral-Inc, [http://www.disruptivetech.com/dt\\_examples/dte-steel.html](http://www.disruptivetech.com/dt_examples/dte-steel.html), date accessed 2002-12-10T11:11:00

ISO, 2000, Management Data (Metadata) for Technical Documents, IEC/ISO 82045, 2000

ISO, 2001, Electronic business eXtensible markup language, ISO15000

JSR, 2004, JSR-000170 Content Repository for Java™ Technology API, Java Community Process, available: <http://jcp.org/aboutJava/communityprocess/review/jsr170/index.html>, date accessed: 20<sup>th</sup> December 2007

Kalay, Y. E. 1999, The Future of CAAD: From Computer-Aided Design to Computer-aided Collaboration, CAAD Futures '99 Conference, 7-8 June 1999, Atlanta, GA.

Krajewski, K., 2001, What's up with ASPs?, Association Management, Volume 53, 2001, p. 38-46.

Lane, B., 2003, The Collaborators, Building Magazine Issue 28, 18<sup>th</sup> July 2003, p. 53

Latham, M. 1994 Constructing the Team, London: HMSO

Berners-Lee, Tim; James Hendler and Ora Lassila (2001). "The Semantic Web". Scientific American Magazine, <http://www.sciam.com/article.cfm?id=the-semantic-web&print=true>

Lewin, K. (1946). Action Research and Minority Problems. In Resolving Social Conflicts. . New York : Harper & Row.

Lima, C., Stephens, J. and Böhms, M., 2003, The bcXML: supporting eCommerce and knowledge management in the construction industry, ITcon Vol. 8, Special Issue eWork and eBusiness , pg. 293-308, <http://www.itcon.org/2003/22>

McCoy, D., Pezzini, M., Natis, Y., Schulte, R., Thompson, J., Lheureux, B., Sinur, J. and Kenney, F. 2003, Hype Cycle for Application Integration and Platform Middleware 2003 (R-20-0814), Gartner, 30<sup>th</sup> May 2003, pp 7.

Metcalf, B. (1995), "Metcalf's Law: A network becomes more valuable as it reaches more users," Infoworld, Oct. 2

Moses (2004), Generic XML Messaging System for Application Integration with Collaborative Systems used by the Construction Industry, MSc Project Report, Loughborough University

NCCTP (2004) The NCCTP Project Exchange Standard, <http://www.ncctp.net/>

NCCTP, (2007), What is the NCCTP?, <http://ncctp.constructingexcellence.org.uk/about.jsp>, date accessed 26 November 2007

OASIS (2004a), Web Services Security: SOAP Message Security V1.0, [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=wss](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss)

OASIS (2004b), Web Services Security: Username Token Profile V1.0, [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=wss](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss)

OASIS (2004c), Web Services Security: X.509 Token Profile V1.0, [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=wss](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss)

PIX, 2004, The Project Information Exchange (PIX) Protocol, The Building Centre Trust, available: <http://www.pixprotocol.org.uk/>, date accessed: 10<sup>th</sup> December 2007

Prinz, W., Löh, H., Pallot, M., Schaffers, H., Skarmeta, A. and Decker, S., 2006, ECOSPACE – Towards an Integrated Collaboration Space for eProfessionals, 2006 International Conference on Collaborative Computing: Networking, Applications and Worksharing, colcom, p39

Tolman, F., Rees, R. and Böhms, M., 2001, Building and Construction Extensible Mark-Up Language (bcXML): The C2B / B2C Scenario, Construction Informatics Digital Library <http://itc.scix.net/>, paper w78-2001-67, available: <http://itc.scix.net/data/works/att/w78-2001-67.content.pdf>, date accessed: 11<sup>th</sup> December 2007

Trombly 2002, The Downside of Web Services: Security Risks, Securities Industry News

W3C 2002, Web Services Architecture Requirements, W3C Working Draft, 19 August 2002, <http://www.w3c.org/TR/2002/WD-wsa-reqs-20020819>

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

Wikipedia, 2007, Research, Wikipedia.org, available: <http://en.wikipedia.org/wiki/Research>, accessed 27 January 2008

Wilkinson, P. 2005, Construction Collaboration Technologies: The Extranet Evolution, Taylor & Francis, UK, p68-87

Woods, M. & Trexler, C. J. (2001). 'Linking interpretive theory to practice: examining an underused research tool in agricultural education', Journal of Agricultural Education, 42(2): 68-78.

WSDP (2004), Java Web Services Development Pack 1.4,

WSE (2004), Microsoft Web Services Enhancements 2.0 SP1,

Yanchar, S. C., Gantt, E. E. & Clay, S. L. (2005). On the Nature of Critical Methodology. Theory & Psychology, Vol. 15, No. 1, 27-50

Yeomans, S., Bouchlaghem, D. and El-Hamalawi, A., 2005, Collaborative Extranet working via multiple construction project extranets, Proceedings of the 3<sup>rd</sup> International Conference on Innovation in Architecture, Engineering and Construction, 15 – 17<sup>th</sup> June 2005, pp 213

Yin, R. K. (2002). *Case Study Research: Design and Methods (3<sup>rd</sup> edition)*. Applied social research method series Volume 5. Sage Publications, California. ISBN 0761925538



**APPENDIX A THE IMPACT OF THE NCCTP DATA  
EXCHANGE STANDARD ON THE PROVIDERS OF  
COLLABORATIVE SOFTWARE TO THE UK  
CONSTRUCTION INDUSTRY**

Moses, S., El-Hamalawi, A., & Hassan, T. 2005. The Impact of the NCCTP Data Exchange Standard on the Providers of Collaborative Software to the UK Construction Industry, Proceedings of the 2005 ASCE International Conference on Computing in Civil Engineering; July 12.15, 2005, Cancun, Mexico. Edited by: Lucio Soibelman and Peña-Mora Feniosky

# THE IMPACT OF THE NCCTP DATA EXCHANGE STANDARD ON THE PROVIDERS OF COLLABORATIVE SOFTWARE TO THE UK CONSTRUCTION INDUSTRY

Scott Moses<sup>1</sup>, Ashraf El-Hamalawi<sup>2</sup> and Tarek Hassan<sup>3</sup>

## ABSTRACT

Since 2003 the bulk of major UK construction project collaborative software providers have been working together to develop standards that will allow for project data to be transferred between vendor applications. Under the umbrella of the Network of Construction Collaboration Technology Providers (NCCTP), an XML standard for project exchange has been developed to address user concerns of stability and choice.

Through extensive discussions between vendors, a series of object classes and their relationships were defined, describing the data commonly held within a collaborative system. Each participating organisation then extended their application to transfer project data to and from the standard. An extensive testing program between the member organisations showed that projects could be successfully transferred from one vendor's system to another.

This paper presents the key aspects of the developed standard, document, user and security sections, and how the model addresses the user communities' requirements, of supplier confidence, data security and transferability. It then examines the impact that the developed standard will have upon vendors who provide collaborative software to the construction industry

Contrasting previous information exchange systems developed for these products, the wide industry support given to the standard ensures that it is applicable throughout all similar products.

## KEYWORDS

NCCTP, Data Exchange Standard, XML, Collaboration Software Tools

---

<sup>1</sup> Research Engineers, Centre for Innovative Construction Engineering, Loughborough University, Leicestershire, LE11 3TU, UK, Phone +44 (0)1628 552013, FAX +44 (0)1628 552001, email [scott.moses@causeway.com](mailto:scott.moses@causeway.com)

<sup>2</sup> Lecturer, Civil and Building Engineering Department, Loughborough University, Leicestershire, LE11 3TU, UK, Phone +44 (0)1509 223206, FAX +44 (0)1509 223981, email [A.El-hamalawi@lboro.ac.uk](mailto:A.El-hamalawi@lboro.ac.uk)

<sup>3</sup> Dr, Senior Lecturer and Director of the European Union Research Group, Civil and Building Engineering Department, Loughborough University, Leicestershire, LE11 3TU, UK, Phone +44 (0)1509 222602, FAX +44 (0)1509 223981, email [T.Hassan@Lboro.ac.uk](mailto:T.Hassan@Lboro.ac.uk)

## INTRODUCTION

Collaborative software tools have become increasingly commonplace amongst larger construction projects over recent years, allowing large amounts of project data to be managed with increasing efficiency. At the core of these systems are the abilities to allow project team members to effectively communicate and exchange data with other team members, allowing real tangible benefits to be realised. However despite the many successes that collaborative technology has brought to the construction industry doubts about the technology and those who provide it still remain amongst many clients. This perception has been fuelled by some well publicised problems of vendors, ceasing to trade, and clients been unable to access the data held on the system. With this negative impression coupled with continuing stories questioning the financial positions of some vendors it is not difficult to see why some organisation are still producing paper backup of drawings, as insurance against the unforeseen when using hosted collaborative services (Milton 2003).

It is in this environment of continuing user concerns that a group of vendors, representing the majority of suppliers to the UK Construction Industry have come together with the aim of increasing the markets confidence in collaborative products. In addition to jointly promoting the benefits of the use of collaborative technology on construction project, they have developed a standard that allows for project data to be transferred between compliant collaborative products.

The project data exchange standard developed by the group of vendors describes projects in terms of a number of classes and their interrelationships. The developed standard builds upon work done on the transfer of documents and associated metadata outlined in the DocLink Specification (Watson & Davoodi 2002), and combines it with data that is stored in Collaborative systems. By industry developing the standard through wide vendor participation the generic model of project information that is stored within a collaborative system can be applied to any project collaborative application used by the construction industry.

The development of the standard will have the impact of increasing the confidence that client organisation can have in collaborative tools by providing a mechanism through which project data can be easily extracted and then resurrected in another system. This increased confidence in collaborative products should help speed up the continuing adoption of collaborative tools for use on projects. The ability to offer the functionality to export project data from the collaborative system will become a requirement that all clients will expect, from their tool supplier. This requirement will drive the adoption of the standard to be implemented by all organisations that trade within this particular marketplace.

The importance of collaborative tools to the aim of improving the overall efficiency and effectiveness of the construction industry is significant. Over the coming years the size of the collaborate marketplace within the UK is expected to continue to grow rapidly with an estimated 95% of large projects 65% of medium projects expected to use the tools by 2007 (Compagina 2003). But this drive towards greater utilisation of the technology must be accompanied by developments and improvements made to collaborative products themselves, if these number are to be realised. Through working together to produce a standard that will enable project data transfer between systems, the collaborative providers

are helping to increase deployment of collaborative products by addressing clients needs over product stability and continued access to their useable data.

The key importance of the collaborative marketplace for the construction industry has become apparent recently with large multinational organisations such as Microsoft targeting it specifically. If these enterprise level solutions are widely deployed over the coming years then a method of transferring live project data to these new systems will be required. The developed standard will be able to facilitate this transfer, allowing potential future client needs to be addressed.

### **THE INDUSTRIAL NEED FOR PROJECT DATA EXCHANGE**

Through the drive to improve the efficiency of the construction process, and to meet the targets set for the UK Construction Industry by a Government Funded Report (Egan 1998). Construction organisations have increasingly been employing project collaboration tools to help manage the entire construction process. This has enabled rapid growth of the collaboration vendor market along with an increase in the capabilities and complexities of the tools that are available. This increase in complexity and the costs associated with the management of continually developing collaborative tools has led to these services being mainly supplied by 3<sup>rd</sup> parties.

Currently a multitude of different collaborative tools are available to be selected by organisations participating in the construction process, offering both opportunities and pitfalls to clients. With the wide diversity of different solutions available clients can usually find a system that they like at a price that they are willing to pay, however is the system going to be reliable over the full duration of the project?

The need for project data exchange can be seen from both the client and provider perspective with each seeking different positive outcomes from a single initial initiative. Vendors, as a group are keen to increase utilisation of collaborative products, by promoting the benefits of their use, and clients seeking to gain these benefits in a secure and predictable environment.

Through the development and implementation of the project data exchange standard the vendor community is seeking to change some of the conceptions that have been propagated about the providers of collaborative tools. The standard is aimed at increasing confidence in the tools by providing a mechanism for the extraction of project data from one system to another should the need arise. Since many of the collaborative solutions that are currently being employed to manage project data were born out of the dot.com era, with the surviving organisations still largely unproven (Krojeviski 2001). Although this perception has diminished somewhat as time has passed since the busting of the dot.com bubble, many ITT (Invitation to Tender) proposals sent out by clients still are seeking to determine the long term viability of the vendor. This coupled with articles within industry publications questioning the financial predicament of some providers (Building 2004), continues to effect confidence in the industry.

In the early days of 3<sup>rd</sup> party vendors supplying collaborative solutions, some major problems arose with the services provided by some companies. With a number of vendors closing down after a few weeks, and others stopping their services without informing their user community (Holden 2001), confidence in collaborative tools industry was damaged.

This coupled with customers not been able to retrieve their data from systems that have ceased to trade has cast a shadow over the long term viability of the wider industry.

Once a solution has been selected and is being used on a project then it is difficult to transfer this project data to another environment, known in the industry as product lock-in. Therefore irrespective of the level of service that is provided the same tool must be used throughout the entire project in all but the most exceptional of circumstances, leading some potential clients to conclude that the level of unknowns outweighs the benefits that the tools usage can bring.

Since the creation of large number of different suppliers of collaborative software solutions appeared in the late 90's respected industry watchers, such as Garner have been predicting wide spread consolidation within the sector. This thinking has been reinforced by Lane (2003) who in his analysis of the collaboration industry expected for the UK market to reduce down to 3 main players and for clients of the vendors leaving to be taken over by the remaining participants.

Increasingly the larger clients have begun to select a single system to help to manage all of their projects in a single environment. These clients having spent large sums of money on systems and training staff on how to use these systems will require the ability to migrate their pre-existing projects from other systems to the one that they have purchased. They will be no longer willing to expend large amounts of money training staff on a new system on a project by project basis and will strive for their own system to be adopted for each project which they participate.

#### **EXISTING STANDARDISED DATA TRANSFER METHODS**

When discussing the issue of the transfer of project data between heterogeneous systems, work can be traced back to Bjork (1993) who first attempted the construction of a conceptual model of documents that could form the heart of a construction document management system. This work was then taken forward to show how document management systems could provide a smooth transition towards computer integrated construction (Turk et al. 1994). Then Rezgui & Cooper (1998) advanced the field further by presenting a migration from document-based to model-based information representation and structure, through looking at the inter workings between different document management systems. However an alternative approach was put forward that moved away from the shared model approach to document management by Hajjar & AbouRizk (2000). The definition of organisation, project and document data based on a common data model but customised to a specific sector of the construction industry. The idea being that the incremental development of specialised modules for each specific sector towards industry wide standardisation facilitates immediate realisation of benefits of construction document management systems. Further attempts were made at an international level to standardise the metadata that should be associated with documents (ISO 2000a) and in particular construction related documents (ISO 2000b), but with individual clients specifying what data they wanted to be stored many diverse document metadata implementations have been created. This client specified document metadata system, coupled with a desire to for speedy database access lead vendors to limit metadata held on objects to that which was required by the client only. Today there is no accepted definitive standard for document metadata existing within the collaborative applications that

are used by the construction industry, with clients specifying their own on a project or enterprise basis. With the wide spread emergence of XML as a data exchange platform, many new XML based standard approaches for the construction industry appeared for data exchange ifcXML (IAI 2001), aecXML (IAI 2004). While aecXML was aimed at producing a collection of transaction schemas, ifcXML did offer storage and transfer capabilities applicable to Collaborative systems. The Leeds University (2002) DocLink specification then extended the ifcXML model and applied this to the transfer of documents and associated metadata between collaborative systems.

### **THE NCCTP STANDARD**

The Standard defines a set of classes and the interrelationships between these classes which represents the commonality that exists between all collaborative systems used by the construction industry. The purpose of the standard is to enable the transfer of a single project, including all associated documents, users and their organisations, from one collaborative system to another. The standard agreed upon by the majority of the providers to the UK construction industry represents the data that is actually stored within the projects, which are held within different provider's systems. As well as agreement upon the core data to be transferred, classes, the standard also represents the structure in which the data will fit, class interrelationships.

The NCCTP Standard document implemented by participating vendors is based upon XML (eXtensible Markup Language), which is an open web standard published by the World Wide Web Consortium (W3C 2000). Since its inception XML has become the paramount standard for data transfer used throughout the world, and has spawned a number of different XML related technologies. The actual standard uses XSD (XML Schema Definition) Language, a Recommendation of the World Wide Web Consortium (W3C 2001), to specify how to formally describe the elements in an XML document. XSD has become the main language for controlling the contents of XML documents rising above its rivals such as the DTD (Document Type Definition) and the XDR (XML Data Reduced) schema backed by Microsoft.

In order to provide support for the standard each participating vendor organisations collaborative application has the ability to export a single project's data to the current version of the NCCTP standard. And also has the able to import project data that is stored in a XML Document that complies with the standard into their own system. In practical terms this means that any future product developments will have to be made with the common standard in mind, so that data can always be moved.

The principles behind the standard are based upon the ability to transfer a single project from one system to another; currently there is no facility to allow for multiple projects to be defined in a single compliant XML document. Since one XML Export document contains one project all organisational and user information relating to this project must be contained within the file. However since systems generally hold multiple projects, and organisations that work on a multitude of these, user and organisation information is held outside of the project, therefore it is a requirement to create them inside the project for the purpose of transfer.

As there is great diversity in the way that different products help in the management of construction projects the standard covers only the core elements that go to make up any collaborative system. The characteristics of these common classes, such as documents, revisions, users and groups, have been designed with extensibility in mind, allowing for providers to include all unique metadata they hold on a particular object.

### KEY CLASSES AND CLASS INTERRELATIONSHIPS

The information that is defined within the NCCTP standard is divided into two main subtypes, the actual documents that exist within the project, and the people who interact with these documents. Within a compliant XML document all definitions of users and the structures in which they are held, organisations and groups, are defined within the two NCCTP elements organisations and groups. Similarly all the information regarding documents is contained within the NCCTP element folders. A Diagram showing the main classes and their relationships can be seen in Figure 1.

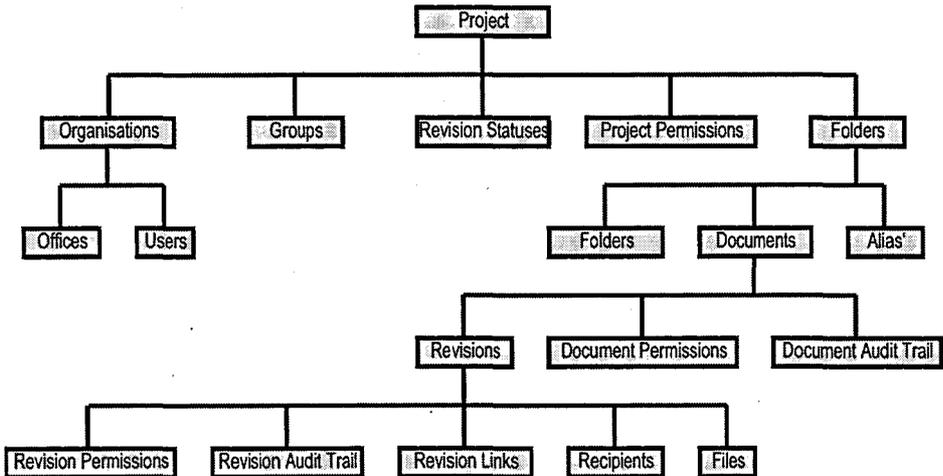


Figure 1: Main Classes and Relationships in the NCCTP Standard

### NCCTP Users

The NCCTP schema holds information about users and organisations that are participants on the project in a separate branch of the XML Document to the folder structure. It defines users and organisations as classes each with their unique set of attributes similar to the ifcXML model for IfcPerson and IfcOrganisation. However a substantial departure from ifcXML is made with the omission of any classes relating to the ifcPersonAndOrganization entity, with relationships between the two classes enforced by having users nested within organisations. Extensions have been made to the classes available in ifcXML to include a group class which

is not included as an ifcXML Class; this follows work done in applying ifcActorResources to project extranets done in the DocLink specifications (Leeds University 2002). A group, as defined by NCCTP standard as being a collection of users, which may or may not be from the same organisation that can act collectively within the collaborative environment. The NCCTP standard does not include any interrelationships between organisations, which are defined by IfcOrganisationRelationship, as this hierarchical organisation structure is not commonly stored within collaborative systems.

### **NCCTP Document Model**

Since one of the primary functions of collaborative systems is to store electronic documents, the IFC Document Model with its scope of managing information about and references to document potentially stored electronically, can form the basis for document transfer. This application of the IFC Document Model was shown when it was used to transfer single documents between collaborative systems during the DocLink project. However the Model does not easily specify a way of arranging a set of documents into a hierarchical folder structure that is commonly deployed within collaborative tools. Since the structure in which documents are stored must be transferred between the two systems, the NCCTP Standard defines the class of 'folder' that stores information about these containers. To allow a full folder structure to be formed folder entities can contain other folder entities and so on to form the entire structure.

The NCCTP schema holds document information within its own class with document attribute information and relationships to other classes defined, comparable to ifcDocumentInformation class. Within the standard the information that is held about the document is extensible so applications can supply any custom attributes that they hold. The ifcDocumentInformation class however defines attributes that are redundant in the NCCTP standard, such as the confidentiality attribute, which is replaced by a new security class, see NCCTP Object Security Implementation.

The NCCTP Standard supports linkages between documents, through the revisionReferences class, with the linkages held being those that exist between different revisions of the document, not the documents themselves. This class forms the same function as the IfcDocumentInformationRelationship but at the revision level not the Document level.

### **NCCTP Object Security Implementation**

As security of data contained within collaborative systems is of fundamental importance to the user community it is of paramount importance that a robust security model is transferred within the XML Document. Therefore the standard has defined a class that can be applied at numerous levels throughout the project. The ACL (Access Control List) class defines the people that have rights to work on a particular object within the system, and can be applied to the project, folder, document, revision, or file. Since all vendors operate slightly different security system the classes needed to be generic enough to allow every system to map security settings.

#### IMPLEMENTATION OF THE STANDARD BY INDUSTRY

In order to be complainant with the NCCTP standard for project data exchange vendor organisations must be able to produce both an accurate export of any project's data, that represents the project to the best level that the schema allows, and to import projects stored in the standard into their system. Organisations are also encouraged to use the extensibility provided in the schema to export addition objects that are not directly implied in the standard. This extended data export gives importing organisations more data to work with in an emergency situation.

Currently 7 vendors have implemented the project data exchange standard, allowing their applications to export data to and then import project data, produced by any product back into their systems. These vendors, who now provide a facility to exchange project data through the NCCTP data exchange standard, account for the majority of tools that are utilised for project collaboration on UK construction projects.

As the standard schema only defines the objects that exist and a structure in which project data can be stored, it cannot control the quality of the data that is outputted to it from any vendor's application. Along with this it also cannot control how that project data that is store is reproduced in the importing system. Unless the project process can accurately reproduce the project the usefulness of the standard will be very limited. Therefore additional checks are required that compare that data that is contained within an exported XML Document with that stored within the application, to ensure that the export XML is an accurate representation. Similarly any vendor who is importing project data from the standard to their application will need to go through a series of tests that verifies that the project created in the system is an accurate representation of that stored in the XML Document. Although direct comparisons between the two systems would seem the logical way of checking that the project data had been reproduced correctly there are a number of limitations that make this approach unfeasible.

- Every collaborative system contains and displays information in a unique way, while in some system it may be easy to compare the two projects, other are sufficiently different to make this task difficult.
- Vendors will not wish to give direct access to products to other collaborative providers as this could provide vital information about the working of their products.
- Exporting system may not exist, to be checked against when the data has been imported into the new system; therefore checks on the data could not be made.
- Without the checking of both the import and export independently any errors that were apparent between the two systems, would not be known where they were generated.

To ensure that the standard that had been developed was capable of enabling project data to be transferred between different applications member organisations of the NCCTP conducted a series of project transfer test. These tests involved the transfer of sample project data between the different vendor applications, and the recreation of the project in the original exporting system. Therefore each vendor produced a XML Document that was

compliant with the data exchange standard of a project stored within their collaborative system. These projects were then made available to other vendor for import into their applications, confirming that the standard could enable project data transfer between different vendor applications.

Not included in the inter-organisational testing were any extensions that individual vendors may have added to their export through the extensibility provided by standard, as this information could give insights in to the product. Or, the transfer of any real complete client project data between systems, as this would be legally impossible because of client confidentiality arrangements.

### **IMPACT ON PROVIDERS OF PROJECT COLLABORATION TOOLS TO THE CONSTRUCTION INDUSTRY**

Since the standard provides a mechanism for the definition of users and organisations that are associated with a particular project, and that organisation may utilise many different collaborative tool providers. Inconsistencies between the information stored in systems will exist. When moving the data from one system to another, these discrepancies could make project transfer difficult through mapping of users and organisations. Currently different collaborative applications required differing amounts data to be compulsorily supplied for any users and organisations, with the primary aim of getting users up and running on the system rather than complete data entry. In addition to this desire to get the users using the system as quickly as possible there are no checks that verify that the data entered is accurate. This will become more important as larger organisations start to employ their own enterprise solutions for project collaboration that will have user information synchronized through directory services. With user uncertainty existing, a great effort will be required in order to make sure that users are mapped correctly, and additional user accounts are not created when the user already exists on the system. In order to better facilitate project data transfer between different systems the collaborative community will be required to overcome this user duplication/confusion problem especially if they are looking to make future advancements into real-time collaboration with users able to access data store in numerous different products.

The standard is a starting point to address some of the other issues that are holding back deployment of collaborative projects, like users being forced to use a particular product, which generates hostility to collaborative products in general meaning that not all the benefits that should be realised are.

The NCCTP standard is not seeking to define a model that all collaborative systems must adopt, that is why only the core is defined. It is enough that systems can map what they store to the collaborative generic model. However if many organisation do a certain item that is seen as a core activity then others may want to adopt this in their products. This will help integration between the differing systems. In real terms this agreed standard means that vendor applications will now not be able to fundamentally change what they do without taking into account project data transform via the schema provided.

The transferring of projects from hosted to enterprise systems and archiving projects from one system and then reinstalling into another to leverage the knowledge that has been developed by the participating vendors.

## CONCLUSIONS

An industry lead initiative has developed an XML based standard to enable the exchange of project data between different collaborative software vendors, working in the UK marketplace. Under the management of CIRIA the NCCTP, who encompass the majority of providers of collaborative software, have defined a set of classes and their interrelationships that model the data commonly held within all collaborative tools. This generic applicability of the constructed data model has been shown through vendors adopting the standard who were not involved in its initial conception ("Business Collaborator joins the NCCTP", 2004), and should reassure other that the standard can work with their applications.

Through the knowledge that developed standards are of little practical use unless widely implemented within industry, participating providers have been proceeding with extensive inter-organisational testing in order to fully implement the standard. With a generic tested solution for the transfer of project data between collaborative systems now in place, amongst the majority of providers to the UK marketplace, users now have the confidence that they are no longer locked into a single solution for the entirety of the project. The implemented standard also reassures users that should the unforeseen happen and the collaborative technology provider go out of business, then the data can be saved and resurrected in another application. The developed exchange standard will allow for the continued deployment of collaborative tools to manage project data, with the knowledge that the information is secure, safe and now potentially transferable should the need arise.

It is not envisaged that clients will be rushing to move live projects between collaborative systems, due to the loss of some application specific data, and that the main application of the standard will probably be to extract project in emergency situations. However, the standard through its common definitions and understandings can form the basis for greater levels of integration between collaborative products in the future. Enabling client's access to all projects data through a single interface, regardless of what collaborative application is storing the information, through multiple collaborative systems collaborating on a single project through real time messaging using standards frameworks such as Web Services.

## ACKNOWLEDGEMENTS

The author wishes to thank the NCCTP organisation and the following providers of collaborative technology who participated in the development of the NCCTP project data exchange standard, 4Projects, BIW, Build Online, Business Collaborator, Cadweb, Causeway Technologies and Sarcophagus.

## REFERENCES

- Björk, B-C., Huovila, P., Hult, S. (1993) Integrated Construction Project Document Management (ICPDM), Proceedings of Europa '93, Delft, June 21-24, pp 135-146.
- Building (2004) Revenue Probes IT Provider BIW, Building Magazine Issue 37
- "Business Collaborator Joins the NCCTP", (2004, February 1), (ncctp.net), Available: [http://www.ncctp.net/news\\_010204.htm](http://www.ncctp.net/news_010204.htm) (Accessed: 2004, November 30).
- Compagina (2003), Collaboration Software in the construction Industry, London.

- Egan, J (1998) Rethinking construction: The report of the construction task force, Department of the Environment, Transport and the Regions, London.
- Hajjar, D. & AbouRizk, S.M. (2000) Integrating Document Management with Project and Company Data, ASCE Journal of Computing in Civil Engineering, January, 14(1), pp 70-77.
- Holden, G. (2001) Outsmarting the Outsourcers, in: Forbes, Special Volume: Best of the Web B2B, 2001, p. 52-53.
- IAI (2001) ifcXML recommended standard, [http://www.iai-international.org/iai\\_international/Technical\\_Documents/IFCXML.htm](http://www.iai-international.org/iai_international/Technical_Documents/IFCXML.htm)
- IAI (2004) aecXML standard documents, <http://www.iai-na.org/aecxml/documents.php>
- ISO (2000a) Management Data (Metadata) for Technical Documents, IEC/ISO 82045, 2000
- ISO (2000b) Technical Product Documentation – Metadata for Construction Documentation, PD ISO 19033, 2000.
- Krajewski, K. (2001) What's up with ASPs?, Association Management, Volume 53, 2001, p. 38-46.
- Lane, B. (2003) Building, 18<sup>th</sup> July 2003, p. 53
- Leeds University (2002) The DocLink Specification, <http://www.doclink.info/>
- Milton, A. (2003). The future direction of project extranets, Project Extranets IV Conference, London, UK, Wednesday 3rd December 2003.
- NCCTP (2004) The NCCTP Project Exchange Standard, <http://www.ncctp.net/>
- Rezgui, Y., Cooper, G. (1998) A Proposed open infrastructure for construction project document sharing, Electronic Journal of Information Technology in Construction (ITcon). 3. pp 11-24.
- Turk, Z., Björk, B-C., Johansson, C., 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.
- W3C (2000) Extensible Markup Language (XML) 1.0 (Second Edition), <http://www.w3.org/tr/rec-xml.html>
- W3C (2001) XML Schema, <http://www.w3.org/xml/schema#dev>
- Watson, A., & Davoodi, M. (2002) Transferring Project Document and Associated Metadata between company Document Management Systems and Project Extranets, eSmart and CISEMIC 2002.

**APPENDIX B THE NCCTP DATA EXCHANGE  
STANDARDS IMPACT ON THE UK  
CONSTRUCTION INDUSTRIES  
COLLABORATIVE TECHNOLOGY USER  
COMMUNITY**

Moses, S., Tarek, H., & El-Hamalawi, A. 2005. The NCCTP Data Exchange Standards Impact on the UK Construction Industries Collaborative User Community, Proceedings of the 3rd International Conference on Innovation in Architecture, Engineering and Construction, Rotterdam, The Netherlands, 15 – 17th June, 2005. Edited by: Sariyildiz, S. & Tunçer, B.

# The NCCTP Data Exchange Standards Impact on the UK Construction Industries Collaborative Technology User Community

Scott Moses\*

*[cvs8@lboro.ac.uk](mailto:cvs8@lboro.ac.uk)*

*Centre for Innovative Construction Engineering, Loughborough University,  
Loughborough, LE11 3TU, United Kingdom.*

Tarek Hassan & Ashraf El-Hamalawi

*Department of Civil and Building Engineering, Loughborough University,  
Loughborough LE11 3TU, United Kingdom.*

## ABSTRACT

Collaborative systems are increasingly being used to manage project information on the larger construction projects. However this speed of expansion coupled with the lack of consolidation between such systems has lead to a highly fragmented marketplace for collaborative products. Currently client organisations are free to select systems from any of the available providers, but once selected are unable to effectively change service provider until the conclusion of the project. This perceived lock-in along with concerns over the stability of some technology providers has created unease amongst the user community and is hindering the adoption of collaborative tools.

In an effort to reassure the users of collaborative tools that these products can be trusted with valuable project data, UK collaborative technology providers to the construction industry have come together to address the user communities concerns. Seven companies agreed to work together on common industry issues. This resulted in the formation of the Network of Construction Collaboration Technology Providers (NCCTP). The NCCTP is a vendor neutral group aimed at increasing interoperability between different collaboration systems. Initially focusing on reducing the concerns associated with the restricted access to project information, an XML data exchange standard was developed. This has been adopted by the NCCTP service providers and allows for project data to be transferred, between different collaboration systems.

This paper examines what the existence of a generic solution for the transfer of project data between collaborative systems will mean for the user community, and why such a standard was of key importance. It introduces the key classes that are defined in the standard, and the interrelationships between them. Moreover how the standard reassures users that, if required, their data can be exported from one system and imported into another application. Also with the current industry trend towards joint ventures, and the increasing prevalence of companies selecting a single collaborative solution, project data transfer will become a necessity in the future. It is the author's belief that the work done on this standard could form the basis of more advanced levels of integration between the different collaborative products.

**Keywords:** Construction, Collaboration Software, Data Exchange Standards, NCCTP, XML Schema

## 1 INTRODUCTION

Collaborative Systems are now an integral part of any major construction project, managing large amounts of documents, facilitating collaboration between project team members and allowing users to quickly find the information required to complete tasks. Modern Collaborative Systems have come a long way since they initially started to be used, with a multitude of different solutions now available for clients to select from. Current offerings range from hosted extranet sites, through to deployments of enterprise level solution to manage all projects. With the complexity of these solutions increasing the initial trend of construction companies deploying their own internal project extranet systems has been replaced with a large number of different vendors supplying collaborative solutions to the construction sector.

The recent rapid growth of the UK collaborative marketplace is expected to continue with an estimated 95% of large projects and 65% of medium projects likely to use the tools by 2007 ("Collaboration Software in the Construction Industry", 2003). These growth figures however will only be realised if vendors continue to address client requirements and concerns, in a timely manner. Since, despite the successes that collaborative technology has brought to the construction industry doubts still remain about the providers sector amongst some clients. With some high profile and well publicised problems with vendors, ceasing to trade and clients been unable to access their data, it is not hard to see why some organisations still are producing paper backup of drawings as insurance against the unforeseen (Milton 2003). It is in this context that a group of vendors, representing the majority of suppliers to the UK Construction Industry have come together with the aim of increasing confidence in collaborative products. In addition to jointly promoting the benefits of the use of collaborative technology on construction project, they have developed a standard that allows for project data to be transferred between compliant collaborative products, in an effort to increase user confidence in the solutions.

The new data exchange standard developed utilises knowledge gained from previous projects conducted on document transfer (Watson & Davoodi 2002), in which one of the NCCTP members was involved, and combines this with knowledge of the actual data that is stored in collaborative systems. By combining a wide sample of vendors who provide services to almost all UK clients an accurate picture of current system usage was created. The constructed generic model of a collaborative system, as actually deployed by clients, was then arranged into a number of classes, with class being used to describe a range of things that have common characteristics. For example in a collaborative system, every folder has the attributes for the date it was created; and every user has attributes of a first and last name. Folder and user are described as names of classes. To continue the analogy there could be many folders used in the project, with each folder conforming to the class specification but potentially having different values assigned to its attributes. One folder may be hidden from view whilst another may be highlighted at the start of the page. Both are folders and have the attributes of a folder as defined in the NCCTP schema. In this respect the work closely resembles the structure of the Industry Foundation Classes defined in ifcXML (IAI 2001) in the concept of defined classes and interrelationships between classes. Indeed many classes are shared between the two standards, with the

ifcXML's being more generic and the NCCTP being more specific to collaborative tool specific.

## 2 THE INDUSTRY NEED

In an effort to meet the targets set out by Egan (1998) the UK construction industry has employed a number of technical solutions to improve its performance, with one of the most successful of these solutions being project collaboration tools. Although initially many organisations developed their own internal collaborative solutions these have largely been replaced by third party systems as the solutions have become more and more elaborate. Thus the collaborative project information management needs for most current construction projects are met by a number of different vendors. This shift towards a service that is provided to the company rather than a resource that the company has direct control over has led to a number of concerns about the reliability of the solutions offered, and the longevity of some vendor' business models. The following list of points illustrates the industrial need for a mechanism allowing the transfer of project data between collaborative systems.

- Once a particular vendor's solution has been selected and is being used on a project, it is difficult to transfer this project data to another environment, known in the industry as product lock-in.
- Many of the collaborative solution that are currently been employed to manage project data were born out of the dot.com era, with the surviving organisations still largely unproven (Krojevski 2001). Although this perception has diminished somewhat as time has passed since the busting of the dot.com bubble, many ITT (Invitation to Tender) proposals sent out by clients still are seeking to determine the long term viability of the vendor. This coupled with articles in industry publications questioning the financial predicament of some providers (Building 2004), continues to effect confidence in the collaboration provider industry.
- In the early years of vendors supplying collaborative solutions, some major problems arose with the services provided by some companies. With a number of vendors closing down after a few weeks, and other stopping their services without informing their user community (Holden 2001), confidence in collaborative tools industry was damaged. This coupled with customers not been able to retrieve their data from systems that have ceased to trade has cast a shadow over the long term viability of the wider industry.
- Since the rapid growth of the sector has led to the creation of a large number of different suppliers of collaborative software solutions industry watchers, such as Garner have been predicting wide spread consolidation. This thinking has been reinforced by Lane (2003) who in his analysis of the collaboration industry expected that the UK market would consolidate to 3 main players and those clients of the vendors leaving would be taken over by the remaining participants.
- Increasingly the larger collaborative technology clients have begun to select a single system to help manage all of their projects in a single environment. These clients having spent large sums of money on systems and training staff on how to use these systems effectively will require the ability to migrate their pre-existing projects from other systems to the one that they have purchased. They will be no longer willing to expend large amounts of money training

staff on a new system on a project by project basis and will strive for their system to be adopted for each project in which they participate.

The standard discussed in this paper addresses the growing user need for a mechanism that will enable project data to be transferred for one vendor's collaborative system, to a selection of other systems, in a way that the active project can continue to be used by the current group of participating users and organisations. Although the solution proposed is only directly relevant to construction project data, the approach of industries participating vendors cooperating could be mirrored in other sectors to leverage similar results, to the benefit of all collaborative users in many vertical markets.

### 3 EXISTING STANDARDISED DATA TRANSFER METHODS

To allow for project data to be transferred between heterogeneous collaborative systems used by the construction industry a conceptual model of the data stored is required. Part of this work was attempted by Bjork (1993) who presented a model of documents that could form the heart of a construction document management system. The early work was then advanced to show how the use of document management systems could provide a smooth transition towards computer integrated construction (Turk et al. 1994). The field was then progressed further by Rezgui & Cooper (1998) who examined the workings of different document management systems, going on to present a migration from document-based to model-based information representation and structure. This however was contradicted when an alternative approach was put forward that moved away from the shared model approach to document management (Hajjar & AbouRizk 2000). With the definition of organisation, document data and project based on a common data model but customised to a specific construction industry sector. The concept being that the incremental development of specialised modules for each specific sector would help move towards industry wide standardisation facilitating the immediate realisation of benefits of construction document management systems. Additional attempts were made at an international level to standardise the metadata associated with documents (ISO 2000a) and in particular construction related documents (ISO 2000b), but with individual clients specifying the metadata they required many diverse document metadata implementations have been created. It is these client specified document metadata systems, coupled with a desire for speedy database access which has lead vendors to limit metadata held to that which is required by the clients only. Today there is no widely accepted definitive standard for document metadata existing within the collaborative applications used by the construction industry, with clients specifying their own on a per-project or enterprise basis.

The wide spread emergence of XML as a data exchange platform, has lead to many new XML based data exchange standards for the construction industry such as ifcXML (IAI 2001) and aecXML (IAI 2004). While aecXML initiative was aimed at producing a collection of transaction schemas, ifcXML did offer storage and transfer capabilities applicable to Collaborative systems. The Leeds University (2002) DocLink specification then extended the ifcXML model and applied this to the transfer of documents and associated metadata between collaborative systems, developing a series of data transfer transactions that could be executed in near real-time.

## **4 THE NCCTP STANDARD**

The Standard defines a set of classes and the interrelationships between these classes which define the commonality that exists between the majority of construction specific collaborative systems used by the construction industry. The purpose of the standard is to enable the transfer of a single project, including all associated documents, users and their organisations, from one collaborative system to another. The standard agreed upon represents the data that is actually stored within projects, which are held within different providers' systems. As well as agreement upon the core data to be transferred, classes, the standard also represents the structure in which the data will fit, class interrelationships.

The NCCTP Standard document implemented by participating vendors is based upon XML (eXtensible Markup Language), which is an open web standard published by the World Wide Web Consortium (W3C 2000). Since its inception XML has become the paramount standard for data transfer used throughout the world, and has spawned a number of different XML related technologies. The actual standard uses XSD (XML Schema Definition) Language, a recommendation of the World Wide Web Consortium (W3C 2001), to specify how to formally describe the elements in an XML document. XSD has become the main language for controlling the contents of XML documents rising above its rivals such as the DTD (Document Type Definition) and the XDR (XML Data Reduced) schema backed by Microsoft.

In order to provide support for the standard each participating vendor organisation's collaborative application has the ability to export a single project's data to the current version of the NCCTP standard. And also has the ability to import project data that is stored in a XML Document that complies with the standard into their own system. In practical terms this means that any future product developments will have to be made with the common standard in mind, so that data can always be moved.

The principles behind the standard are based upon the ability to transfer a single project from one system to another; so currently there is no facility to allow for multiple projects to be defined in a single compliant XML document. Since one XML Export document contains one project all organisational and user information relating to this project must be contained within the file. However since systems generally hold multiple projects, and organisations work on a multitude of these, user and organisation information is held outside of the project, therefore it is a requirement to create them inside the project for the purpose of transfer.

As great diversity exists in the way that different products help in the management of construction projects the standard covers only the core elements that go to make up any collaborative system. The characteristics of these common classes, such as documents, revisions, users and groups, have been designed with extensibility in mind, allowing for providers to include all unique metadata they hold on a particular object.

### **4.1 Key Classes and Class Interrelationships**

The Information that is defined within the NCCTP standard is divided into two main subtypes, the actual documents that exist within the project, and the people who interact with these documents. Within a compliant XML document all definitions of

users and the structures in which they are held, organisations and groups, are defined within the two NCCTP elements organisations and groups. Similarly all the information regarding documents is contained within the NCCTP element folders. A Diagram showing the main classes and their relationships can be seen in Figure 1.

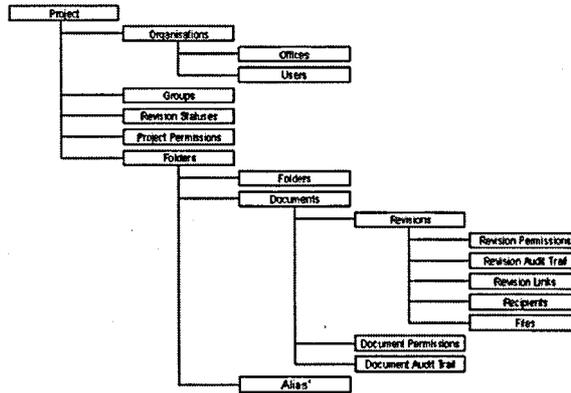


Figure 1: Main Classes and Relationships in the NCCTP Standard

### NCCTP Users

The NCCTP schema defines users and organisations as classes each with their unique set of attributes similar conceptually to the ifcXML model for IfcPerson and IfcOrganisation. However a substantial departure from ifcXML is made with the omission of any classes relating to the ifcPersonAndOrganization entity, with relationships between the two classes enforced by having users nested within organisations. Extensions have been made to the classes available in ifcXML to include a group class; this follows work done in applying ifcActorResources to project extranets done in the DocLink specifications (Leeds University 2002). A group, is defined by NCCTP standard as being a collection of users, which may or may not be from the same organisation that can act collectively within the collaborative environment. The NCCTP standard does not include any interrelationships between organisations, which are defined by IfcOrganisationRelationship, as this hierarchical organisation structure is usually not stored within collaborative systems.

### NCCTP Document Model

As a primary function of collaborative systems is to store electronic documents, the IFC Document Model with its scope of managing information about and references to documents potentially stored electronically, can form the basis for document transfer. This application of the IFC Document Model was shown when it was used to transfer single documents between collaborative systems during the DocLink project. However the Model does not specify a way of arranging a set of documents into a hierarchical folder structure that is commonly deployed within collaborative tools. Since the structure in which documents are stored must be transferred between the two systems, the NCCTP Standard defines the class of 'folder' that stores information about these containers. To allow a full folder structure to be formed folder entities can contain other folder entities and so on to form the entire structure.

The NCCTP schema holds document information within its own class with document attribute information and relationships to other classes defined, comparable to `ifcDocumentInformation` class. Within the standard the information that is held about the document is extensible so applications can supply any custom attributes that they hold. The `ifcDcoumentInformation` class however defines attributes that are redundant in the NCCTP standard, such as the confidentiality attribute, which is replaced by a new security class, see NCCTP Object Security Implementation.

The NCCTP Standard supports a mechanism for describing relationships between revisions of different documents, through the `revisionReferences` class. This class forms the same function as the `IfcDocumentInformationRelationship` but at the document revision level not the document level.

### NCCTP Object Security Implementation

As security of data contained within collaborative systems is of fundamental importance to the user community it is of paramount importance that a robust security model is transferred within the XML Document. Therefore the standard has defined a class that can be applied at numerous levels throughout the project. The ACL (Access Control List) class defines the people that have rights to work on a particular object within the system, and can be applied to the project, folder, document, revision, or file. Since all vendors operate slightly different security system the classes needed to be generic enough to allow every system to map security settings.

### 4.2 Implementation of the Standard by Industry

The standard has so far been implemented by 7 different organisations that account for the lion share of tools that are currently been used on UK construction projects, table 1. Although most of these organisation had input into the initial drafting of the standard one vendor joined after the first version of the standard had been agreed upon, but had no problems mapping their system to the NCCTP standard.

NCCTP Member Organisation
4Projects
BIW
Build Online
Business Collaborator
Cadweb
Causeway Technologies
Sarcophagus

*Table 1: Collaborative Technology providers involved in the standards development*

To ensure that project data could be successfully transferred between the participating systems, a series of test projects were created in each system, and then exported to the standard. The validity of the created XML data was verified through the conduction of a series of standard export tests defined by the NCCTP. These exported XML documents were then made available for each different organisations application to import into their system. Once that the data had been imported into a system the validity of the data was verified through a series of standard import tests defined by the NCCTP. The project data transfer testing that was conducted between all the

member organisations showed that data could be successfully moved between systems.

The standard outlines a structure in which project data can be held that is neutral of any one organisation system. When the standard was implemented by the different vendor organisations no organisation changed the way that they held the data within their own database system. Indeed the aim of the exchange standard is not to change how data is stored, only to provide a mechanism for the storage of project data that can be understood by all vendors.

Currently no clients have utilised the standard through choice or necessity, to transfer a live project between different systems, with the main client interest in archiving completed projects and recreating them in enterprise systems to leverage the knowledge gathered.

## **5 USER IMPACT OF NCCTP STANDARD**

For the first time users of any NCCTP compliant system will have the ability to take information relating to a 'live' project from one system and transfer it to another system, without the need for a lengthy data mapping exercise to be carried out by the two vendors involved. Furthermore as the system has been implemented by a large number of vendors, the client organisation has the ability to select a system that best meets their requirements. For example if the system in which the data was originally stored employed a hierarchical folder structure, similar to windows explorer, then it is likely that the client would wish to import project data into a system that used hierarchical folders. The ability for users to access their data in XML format is nothing new, with the majority of systems already offering XML Export functionality to their clients. However the standard devised by NCCTP members allows project information to be exported in such a way that another application can readily interpret the data and recreate the project in another system.

The emergence of the standard may have little impact on the system selection decision made by vendors but it should increase the confidence to use this technology. Users if collaboration systems are likely to extend their use of such tools when they are confident that increasing the level of intellectual property contained in such systems will remain accessible. Such accessibility is partly to address concerns should a vendor cease trading but more about the ability to respond to different business requirements – such as working with clients and partners who may use an alternative collaboration system.

The close cooperation between the different vendors and the production of a standard will be of benefit to the clients of such applications. This also shows that vendors can work together for the common good of the industry. The industry made the need for action clear and will continue to do so as their business requirements evolve over time. The formation of the NCCTP creates a focus for users and providers alike. This collaboration in partnership with the user community should be to the benefit of the whole industry.

## 6 CONCLUSIONS AND FUTURE DEVELOPMENTS

With the majority of providers of collaborative software to the UK construction industry participating and implementing the data exchange standard. It will become standard functionality that clients should expect from any supplier that they engage with. As the development of the standard has been industry lead, and involved a large number of the main providers it cannot be seen as been biased to one particular vendor or group of vendor. Other Collaborative providers should feel confident that their product could conform to the standard with has been reinforced by a vendor joining and implementing the full standard after the first version was constructed ("Business Collaborator joins the NCCTP", 2004).

Work by each individual member organisation in implementing that data exchange standard with their individual collaborative tool has started the providers collectively down the road to greater integration in the future. With a generic tested solution for the transfer of project data between collaborative systems now in place, amongst the majority of providers to the UK marketplace, users now have the confidence that they are no longer locked into a single solution for the entirety of the project. The implemented standard also reassures users that should the unforeseen happen and their collaborative technology provider goes out of business, then the data can be saved and resurrected in another application. The developed exchange standard will allow for the continued deployment of collaborative tools to manage project data, with the knowledge that the information is secure, safe and now potentially transferable should the need arise.

The trends towards joint ventures, and the increasing use of enterprise collaborative solution, will mean that users will potentially soon require the ability to access project data stored in numerous different systems. It is the author's belief that work done on this standard could form the basis for this level of real-time integration.

## REFERENCES

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

Building (2004) Revenue Probes IT Provider BIW, Building Magazine Issue 37, 24<sup>th</sup> September 2004, p 14.

"Business Collaborator Joins the NCCTP", (2004, February 1), (ncctp.net), Available: [http://www.ncctp.net/news\\_010204.htm](http://www.ncctp.net/news_010204.htm) (Accessed: 2004, November 30).

"Collaboration Software in the construction Industry", (2003), (Compagna.biz), Available: [http://www.compagnia.biz/h/n/ECONSTRUCTION/proj\\_coll/ALL/43/](http://www.compagnia.biz/h/n/ECONSTRUCTION/proj_coll/ALL/43/) (Accessed: 2003, November 20).

Egan, J (1998) Rethinking construction: The report of the construction task force, Department of the Environment, Transport and the Regions, London. p 16.

Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

---

Hajjar, D. & AbouRizk, S.M. (2000) Integrating Document Management with Project and Company Data, ASCE Journal of Computing in Civil Engineering, January, 14(1), pp 70-77.

Holden, G. (2001) Outsmarting the Outsourcers, in: Forbes, Special Volume: Best of the Web B2B, 2001, pp 52-53.

IAI (2001) ifcXML recommended standard, [http://www.iai-international.org/iai\\_international/Technical\\_Documents/IFCXML.htm](http://www.iai-international.org/iai_international/Technical_Documents/IFCXML.htm)

IAI (2004) aecXML standard documents, <http://www.iai-na.org/aecxml/documents.php>

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

ISO (2000b) Technical Product Documentation – Metadata for Construction Documentation, PD ISO 19033, 2000.

Krajewski, K. (2001) What's up with ASPs?, Association Management, Volume 53, 2001, p. 38-46.

Lane, B. (2003), The Collaborators, Building Magazine Issue 28, 18<sup>th</sup> July 2003, p. 53

Leeds University (2002) The DocLink Specification, <http://www.doclink.info/>

Milton, A. (2003). The future direction of project extranets, Project Extranets IV Conference, London, UK, Wednesday 3rd December 2003.

NCCTP (2004) The NCCTP Project Exchange Standard, <http://www.ncctp.net/>

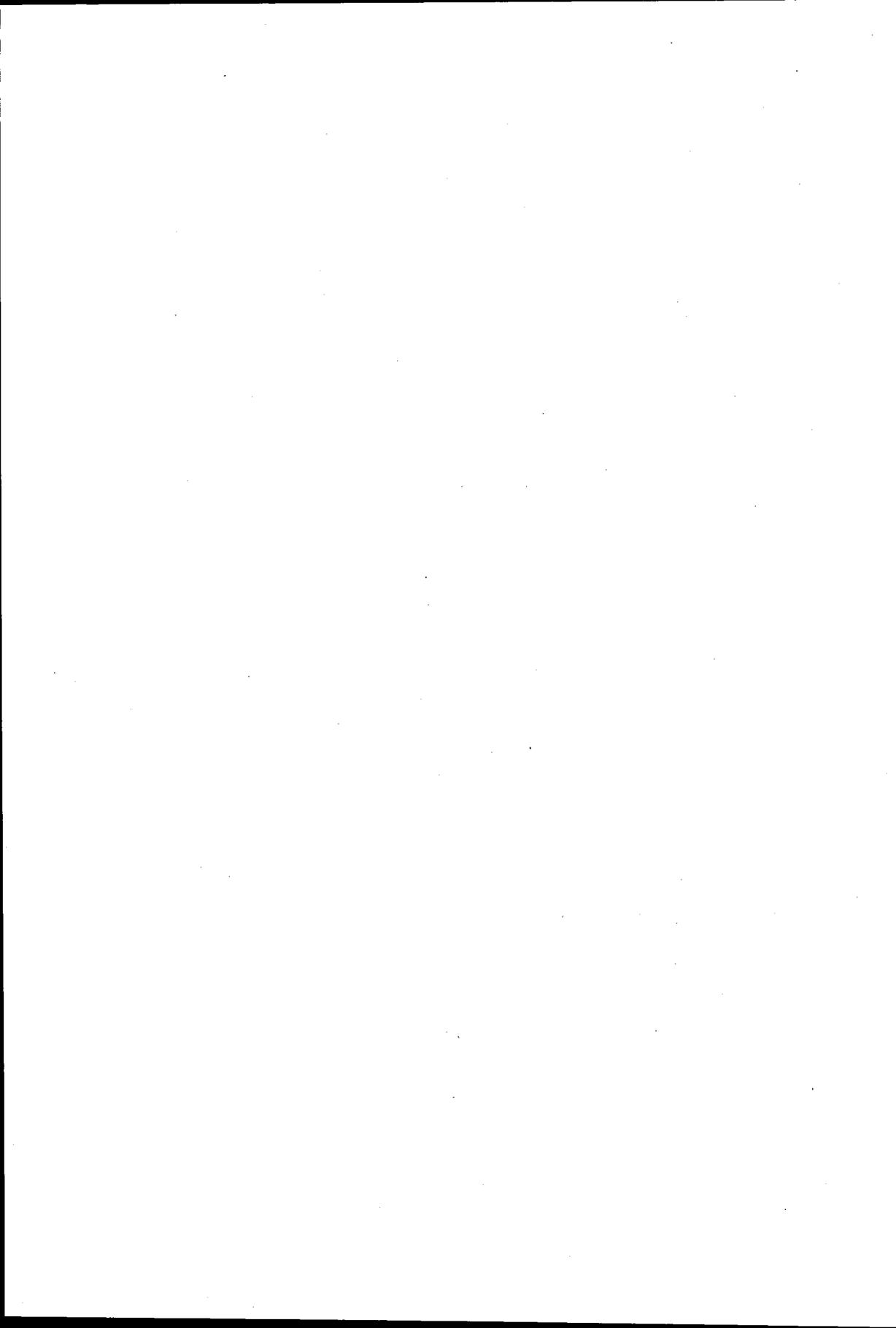
Rezgui, Y., Cooper, G. (1998) A Proposed open infrastructure for construction project document sharing, Electronic Journal of Information Technology in Construction (ITcon). 3. pp 11-24.

Turk, Z., Björk, B-C., Johansson, C., 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.

W3C (2000) Extensible Markup Language (XML) 1.0 (Second Edition), <http://www.w3.org/tr/rec-xml.html>

W3C (2001) XML Schema, <http://www.w3.org/xml/schema#dev>

Watson, A., & Davoodi, M. (2002) Transferring Project Document and Associated Metadata between company Document Management Systems and Project Extranets, eSM@RT conference, Salford, UK, November 2002



**APPENDIX C THE PRACTICALITIES OF  
TRANSFERRING DATA BETWEEN PROJECT  
COLLABORATION SYSTEMS USED BY THE  
CONSTRUCTION INDUSTRY**

Moses, S., El-Hamalawi, A., & Hassan, T. 2007. The Practicalities of Transferring  
Data between Project Collaboration systems used by the Construction Industry,  
Automation in Construction 17 (2008) 824-830

# THE PRACTICALITIES OF TRANSFERRING DATA BETWEEN PROJECT COLLABORATION SYSTEMS USED BY THE CONSTRUCTION INDUSTRY

S. Moses <sup>a,\*</sup>, A. El-Hamalawi <sup>b</sup>, T.M. Hassan <sup>b</sup>

<sup>a</sup> CICE (Centre for Innovative Construction Engineering) Civil and Building Engineering Department, Loughborough University, Loughborough LE11 3TU, UK

<sup>b</sup> 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.

KEYWORDS: XML, project collaboration, data exchange standards, architecture engineering and construction

## 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 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 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. The 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 best 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'/'archived' effectively making it read only, other 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 objects metadata or any object deletions cannot easily be 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 check 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 how the vendor to standard mapping information is used to take data extracted XML data from the source system and reproduce it correctly in the destination system. 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 a generic construction based collaborative system model for individual vendors to map to is the latest in a number of proposals [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 than 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. 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 through 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.

## **10. CASE STUDIES**

### **10.1 TRANSFERRING PROJECT DATA FROM A PROJECT EXTRANET TO A ENTERPRISE COLLABORATION SYSTEM**

Since 2004 HBG, one of the UK's leading construction services organisations, have run an internal collaboration system based upon 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 was 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 for transfer 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, 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.

## **10.2 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.

- 7) Project data was exported from the pilot server and stored in an NCCTP based XML format.
- 8) Exported project data was validated, e.g. checking number of documents, revisions, etc.

- 9) Project data was moved from the pilot server to the live server.
- 10) User account mapping between the pilot and live server was done.
- 11) Project data was imported into the live server.
- 12) 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.

## **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 participant's 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](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 Impact on the UK Construction Industries Collaborative Technology User Community, Proceedings of the 3<sup>rd</sup> International Conference on Innovation in Architecture, Engineering and Construction, Rotterdam, The Netherlands, 15 – 17<sup>th</sup> 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](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 (eXML), ISO15000, 2001
- [14] IAI, 2004, aecXML Standard Documents, available from <http://www.iaina.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-til.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

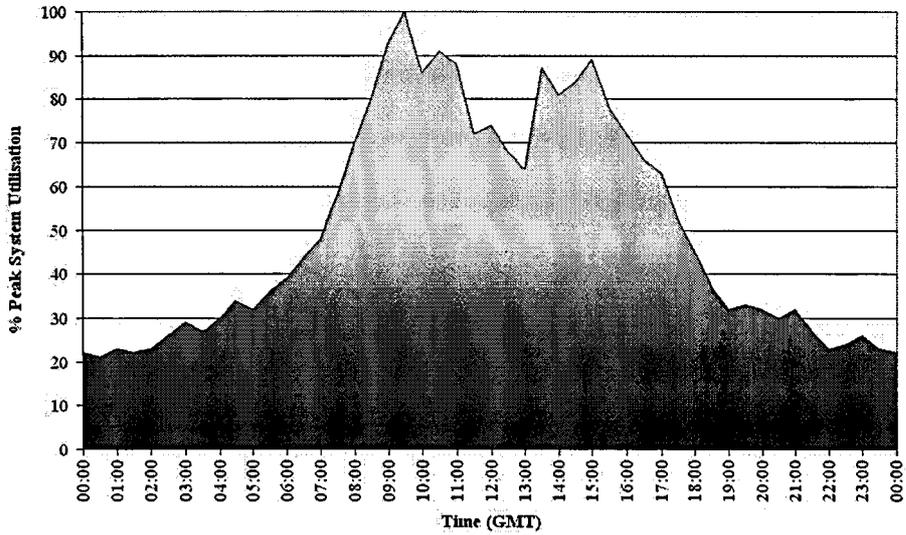


Figure 1 - System Utilisation for UK focused provider with projects throughout the world

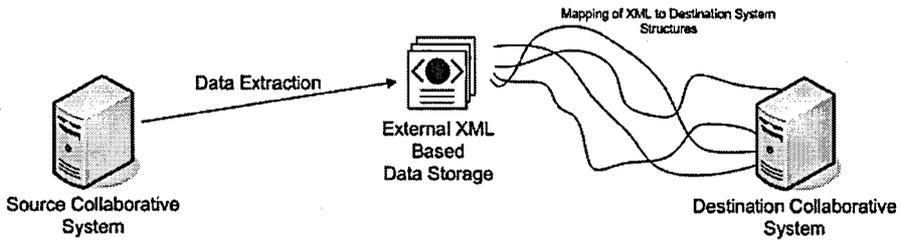
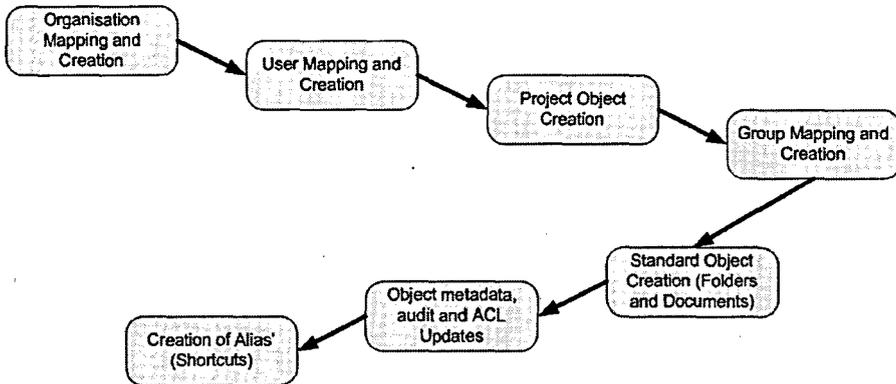


Figure 2 - Mapping Extracted Data to Destination System



**Figure 3 - Standard Procedure for the import of project data**

**Table 5 - Data Security Methodologies**

Methodology	Advantages	Disadvantages
Data Encryption at Generation	No unsecured data is stored outside of either system. Data cannot be modified.	Verification of exported data made more difficult. Extra time needed during the export process. Verification of imported data made more difficult
Data Encryption at Transfer	The data is secured while it is in transit between the two systems. No extra time required at data generation. Verification of data is possible, and quicker to carry out	Data can be changed either after export or before import by the respective extranet systems. Additional Time incurred for the total transfer process.
Data Unencrypted	Verification of data possible both after export and before import. No extra time required to encrypt and decrypt, allowing projects to be live as quickly as possible.	Data can be changed at both ends without anybody knowing. Data can be changed by third parties without the importer or exporter knowing.

**Table 2 – Metrics of the selected HBG Project**

Project Metric	Value
Number of Users	115
Number of Folders	913
Number of Documents	2231
Number of Revisions	3703
Physical Size of Revisions	667MB

Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

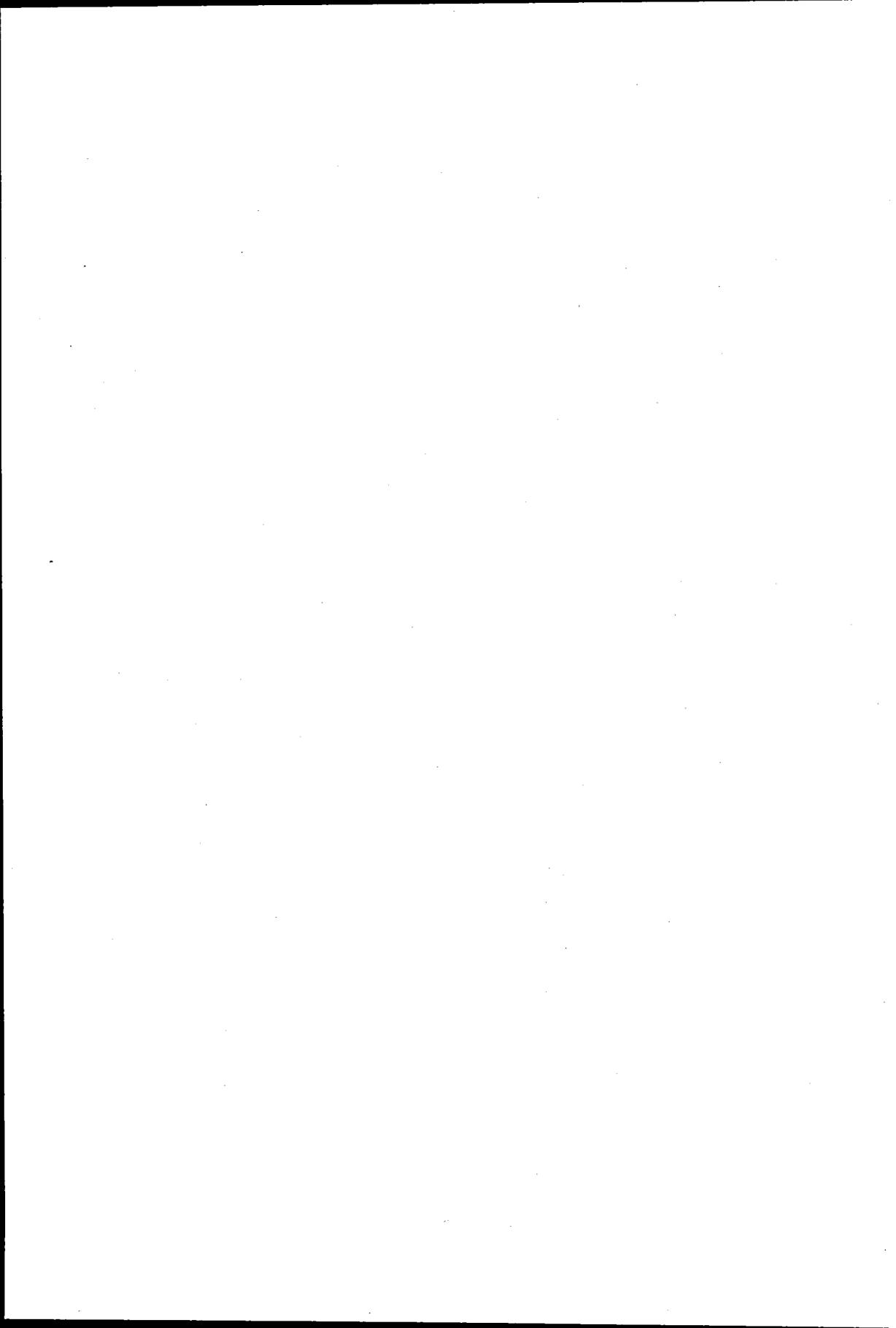
---

**Table 3 – Metrics of the 3 Kier Projects**

<b>Project Metric</b>	<b>#1</b>	<b>#2</b>	<b>#3</b>
Number of Users	86	86	86
Number of Folders	1506	435	432
Number of Emails	912	919	294
Number of Documents	6634	2088	1234
Number of Revisions	7213	2259	1365
Total Size of Revisions	2.32GB	305MB	249MB

**Table 4 – Duration of the Separate Project Migration Activities**

<b>Activity</b>	<b>#1</b>	<b>#2</b>	<b>#3</b>
Export Project Data	75 min	30 min	15 min
Validate Exported Data	20 min	20 min	20 min
Transfer Data	15 min	7 min	5 min
User Account Mapping	60 min	(same users as #1)	(same users as #1)
Import Project Data	153 min	60 min	35 min
Validate Imported Data	20 min	20 min	20 min



**APPENDIX D AN XML BASED STANDARD TO  
ENABLE BULK PROJECT DATA TRANSFER  
BETWEEN HETEROGENEOUS SYSTEMS**

Moses, S., Tarek, H., & El-Hamalawi, A. 2008. An XML Based Standard to Enable  
Bulk Project Data Transfer Between Heterogeneous Systems, ITCon, Vol. XX, No.  
XX, pp. XX – XX. (ACCEPTED FOR PUBLICATION).

# AN XML STANDARD TO ENABLE BULK PROJECT DATA TRANSFER BETWEEN HETEROGENEOUS SYSTEMS

*Scott Moses, Research Engineer  
Centre for Innovative Construction Engineering, Loughborough University  
email: scott.moses@causeway.com*

*Tarek M Hassan, Senior Lecturer and Director of the European Union Research Group  
Department of Civil and Building Engineering, Loughborough University, United Kingdom  
email: T.Hassan@Lboro.ac.uk*

*Ashraf El-Hamalawi, Senior Lecturer  
Department of Civil and Building Engineering, Loughborough University, United Kingdom  
email: A.El-hamalawi@Lboro.ac.uk*

**SUMMARY:** *The objective of the work reported in this paper was to develop an XML standard which would enable project data to be transferred between collaborative systems provided by different software vendors and used by construction organisations. This paper discusses the specific needs for such a transfer capability between collaborative systems within this sector, by examining the current use of these tools and problems encountered by clients. It sets out the main components that underlie the majority of construction specific collaborative systems which forms the basis of the generic collaborative system model which has been developed, contrasting this with previous data exchange efforts. This paper shows how the standard was developed and the procedures undertaken to ensure that it could be utilised by the maximum possible set of vendors. It sets out the best practice procedure for implementations by vendor organisations and the required testing to confirm a successful transfer. The paper also highlights some of the practical problems that were encountered when transferring projects between heterogeneous systems during the project and in subsequent deployments of the solution. Finally, the paper concludes with methods of taking the work forward as a foundation to allow for greater interoperability between systems in the future.*

**KEYWORDS: DATA EXCHANGE, XML SCHEMA, CONSTRUCTION, COLLABORATIVE SOFTWARE, NCCTP**

## 1. INTRODUCTION

The Network of Construction Collaboration Technology Providers (NCCTP) is an organisation of vendors, TABLE 6, founded in 2003, which provides collaborative solutions to construction organisations primarily in the UK, although member's solutions are used on projects throughout the world (NCCTP, 2004). Collaborative systems provided by participating organisations range from project extranets that can be utilised on a single project basis, through to enterprise wide systems with several organisations providing both. Initially containing seven vendors the NCCTP's first project was to develop a vendor neutral standard enabling the transfer of project data form one system to another, the outcomes of which are reported here. Since its initial inception three additional vendor organisations have joined and are participating on

## Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

the continuing efforts towards greater standardisation within the industry, (Aconex, 2005, BCL, 2004, NCCTP, 2005).

**TABLE 6: NCCTP Member Organisations**

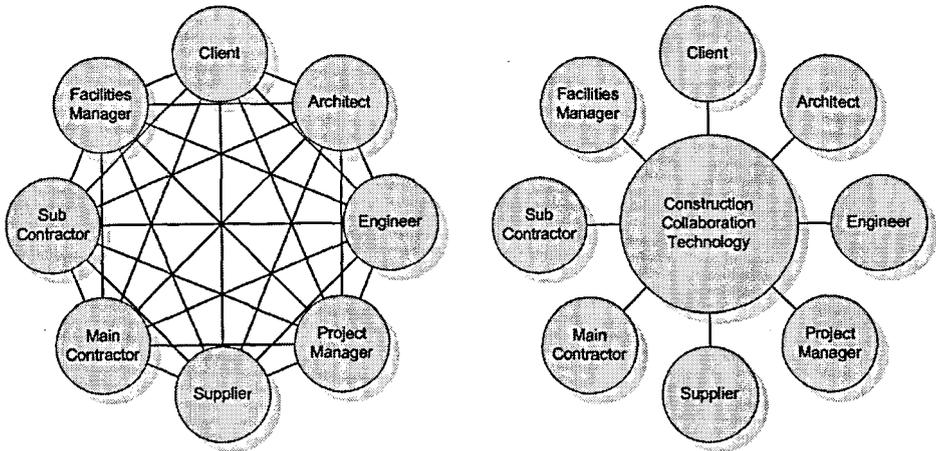
Vendor	Vendor's link to NCCTP
4Projects	<a href="http://www.4projects.com/">http://www.4projects.com/</a>
Aconex	<a href="http://www.aconex.com/Corporate/News/Latest-News/Aconex-Joins-The-NCCTP.html">http://www.aconex.com/Corporate/News/Latest-News/Aconex-Joins-The-NCCTP.html</a>
Asite	<a href="http://www.asite.com/partners.shtml">http://www.asite.com/partners.shtml</a>
BIW	<a href="http://www.biwtech.com/cp_root/h/Media_Centre/BIW_helps_found_NCCTP/272/">http://www.biwtech.com/cp_root/h/Media_Centre/BIW_helps_found_NCCTP/272/</a>
Business Collaborator	<a href="http://www.groupbc.com/">http://www.groupbc.com/</a>
CadWeb	<a href="http://www.cadweb.co.uk/aboutus/?partners">http://www.cadweb.co.uk/aboutus/?partners</a>
Causeway Technologies	<a href="http://www.causeway.com/corpooverview/affiliations.htm">http://www.causeway.com/corpooverview/affiliations.htm</a>
Sarcophagus	<a href="http://www.sarcophagus.co.uk/">http://www.sarcophagus.co.uk/</a>

The bulk project data exchange standard described in this paper approached the challenge from the perspective of the current structure of the deployed solution used on construction projects. By fitting the proposal to the existing systems, rather than trying to define an ideal model of a collaborative system it would be easier for vendors to implement with their current and future products. This approach contrasts earlier interoperability efforts where the proposed solutions were either arrived at from a solely theoretical view point or through collaboration with a single vendor.

This paper presents the work done in creating the NCCTP bulk data exchange standard and its implementation by collaborative system vendors. It sets the context of this work by initially describing the current collaborative systems used by the construction sector, and then examining the industrial need for the bulk project data exchange. It then examines the previous work done in defining a generic model of a collaborative system and how this relates to the model constructed by the NCCTP, for data transfer. The paper then shows the stages involved in transferring project data from one collaborative system to another, which were discovered through inter-organisational testing, followed by the potential barriers which exist to transfer. Finally the paper suggests possible future work which could build on the research presented in this paper.

## 2. CURRENT COLLABORATIVE SYSTEMS

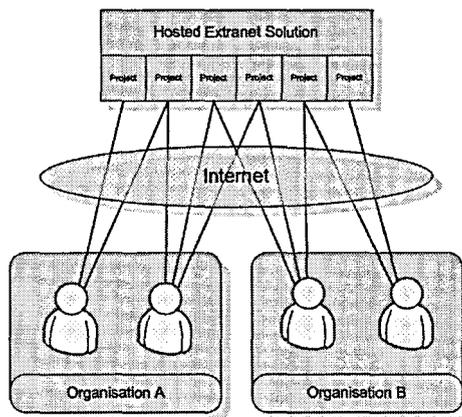
Collaborative systems in general refer to a set of web based technologies which enable employees from a number of different organisations to share information effectively while working on a construction project or a series of projects. *FIG. 1* from (Wilkinson, 2005) shows how collaborative systems have changed the way project participants communicate during the project. These technologies are primarily document management, version control, workflows, drawing management, viewing, mark-up of drawings, searching and permission based access to content. Collaborative systems are known for their high availability and 24/7 access to information, allowing construction organisations to work effectively on projects located throughout the world. Since their inception as hosted project extranets they have expanded into a number of different forms, each of which is best suited for a particular situation, including enterprise and hosted enterprise varieties which are discussed below.



**FIG. 1: Traditional Project Communication (left), and Project Communication using Collaborative System (right)**

## 2.1 HOSTED EXTRANET SOLUTIONS

Hosted extranet solutions are collaborative environments where project data is held in a remote secure location by the actual software vendor, with all infrastructure and applications managed by the vendor. FIG. 2 illustrates this model of multiple self contained projects being accessed by multiple clients over the internet. Normally deployed on a per project basis some bulk agreements do exist, especially with clients who do a large number of construction projects (4Projects & Tesco Express, 2003). Its independence of any of the parties which are collaborating on the project is a major advantage of this type of collaborative solution. However by their nature extranet solutions are usually shared amongst a large number of different companies working on different projects, which can affect performance at peak utilisation times.



**FIG. 2: Hosted Extranet Solution**

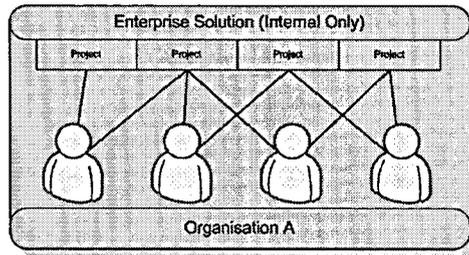
## 2.2 ENTERPRISE SOLUTIONS

Enterprise solutions are collaborative environments that are operated by the construction organisation, with the software provided by a collaborative vendor but

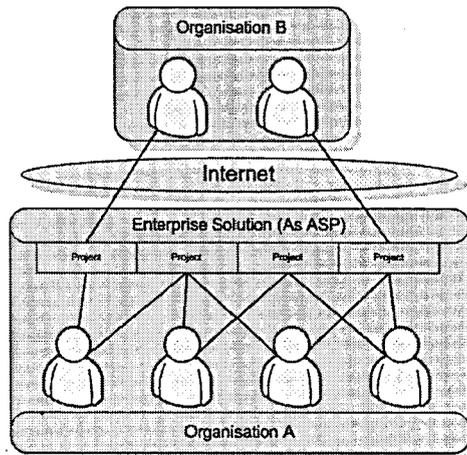
## Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

managed by staff who report directly to the organisation, with examples like Buzzsaw Enterprise Server (AutoDesk, 2004) and Enterprise Collaboration from Business Collaborator (BCL, 2007). These systems allow for an unlimited number of projects and are more easily integrated into the companies other existing systems. Since the system belongs to the client it can be customised to their exact specifications which are not possible when using a single shared extranet solution. Advantages which this flexibility can bring are highlighted by ARUP who deploy Causeway's Collaborative solution as their own ASP service called ARUPLink (Cutler 2006). However since they are operated by a particular organisation, they are not seen as being as independent as hosted solutions.

Enterprise solutions deployed by clients fall into two main categories, those just used by a company's own employees, FIG. 3 and those shared with 3<sup>rd</sup> party organisations, FIG. 4. When an organisation uses an enterprise solution for only internal staff they usually work with a number of hosted extranet solutions as well, to enable collaboration between their organisation and other project participants.



**FIG. 3: Internal Only Enterprise System**



**FIG. 4: Shared Enterprise System**

## 6.1.1

### 2.3 HOSTED ENTERPRISE SOLUTIONS

Hosted enterprise solutions are collaborative environments that are operated by the collaborative software vendor for the sole use of a single client or a single project. The service is managed and maintained by the vendor similar to an extranet system, but since it is single use, it can be customised according to the client or projects' individual requirements. They differ from project extranets as they do not require 24/7 availability, but only that which is required by the customer. However, they are vastly more expensive than their shared equivalents as software and hardware needs to be purchased, unlike extranets where these costs are shared amongst all users.

### 3. THE INDUSTRIAL NEED FOR THE ABILITY TO TRANSFER PROJECT DATA BETWEEN COLLABORATIVE SYSTEMS

The need for a mechanism to bulk transfer project data between different systems can be seen from both the perspective of client and vendor with each seeking different positive outcomes from the transfer ability. Vendors are collectively keen to increase the utilisation of collaborative products, and clients are seeking to gain these benefits in a secure and predictable environment. Work by (Birkby and Nugent, 2002) suggests that transfer provision should be included in the contracts between vendors and clients. The following two sections examine in more detail the need from each stakeholder group.

#### 3.1 THE VENDOR'S NEED FOR DATA TRANSFER

Through the implementation of a project data exchange standard, the vendor community is collectively seeking to change some of the perceptions held by clients about the providers of collaborative tools, by addressing the concern raised by authors e.g. (Birkby and Nugent, 2002, Hampton, 2001, McBride, 2003, Berning and Flanagan, 2003). The standard is aimed at increasing confidence in the tools by providing a mechanism for the extraction of project data from one system and easy importation into another should the need arise.

Increasingly the larger clients have begun to select a single system to help to manage all of their projects in a single environment. These clients having spent large sums of money on systems and training staff on how to use these systems will require the ability to migrate their pre-existing projects from other systems to the one that they have purchased. This barrier to use of collaborative systems by clients was highlighted by the (NCCTP, 2006) 'Proving Collaboration Pays' Survey where 67% of respondents identified training on different tool a significant barrier to use. Further work (Yeomans et al, 2005) had found that the reluctance to retrain for new tools has led to difficulties for system users, creating a negative perception of collaborative tools.

#### 3.2 THE CLIENT'S NEED FOR DATA TRANSFER

In an attempt to meet the improvement targets demanded by Egan, (Egan, 1998) the UK Construction sector has increasingly been employing project extranet technology to manage the vast amount of information generated during a project. In the initial

stages of this drive many construction organisations developed their own solutions, but as the complexity of solutions grew, these were largely replaced by offerings from specialist software vendors. The move towards utilising a service provided by others raised a number of concerns about the reliability of the solutions and the business models operated by the vendors.

Construction organisations were concerned that once a particular service had been selected for a project, they were effectively locked in as it would be exceedingly difficult to get the data out into any other system. Additionally when the concept of enabling data transfer was first discussed, many of the vendors of collaborative software were still largely unproven (Krojevski, 2001). This coupled with industry publications questioning the financial predicament of some providers (Building, 2004), continues to affect confidence in the collaboration provider industry. Client's confidence in collaborative products has been further shaken by some early high profile failures of providers, and then their inability to access their data (Holden, 2001).

Since the rapid growth of the sector has led to the creation of a large number of different suppliers of collaborative software solutions, industry watchers, such as Garner have been predicting wide spread consolidation. This thinking has been reinforced by Lane (Lane, 2003) who in his analysis of the collaboration industry expected that the UK collaborative software market would consolidate to around 3 main vendors. Those vendors leaving the marketplace would need to transfer their remaining active client projects to the vendors continuing to operate.

#### **4. GENERIC COLLABORATIVE SYSTEM MODEL**

Enabling Project Data transfer between heterogeneous systems requires a conceptual model of how data is structured. (Bjork, 1993) presented a model of how documents were stored in such a system. (Rezgui and Cooper, 1998) then progressed the subject further by examining the workings of different document management systems, going on to present a migration from document-based to model-based information representation and structure. However this was contradicted by (Hajjar and AbouRizk, 2000) who put forward an alternative approach which moved away from the shared model approach to document management. Their work on a definition of organisation, document data and project based data specifically customised for the construction sector. (ISO, 2000a) attempted to standardise the metadata associated with documents and (ISO, 2000b) focused on construction related documents. Most of these attempts to standardise the data however were not flexible enough for data transfer needs.

The wide spread emergence of XML as a data exchange platform, has led to many new XML based data exchange standards for the construction industry such as ifcXML (IAI, 2001) and aecXML (IAI, 2004). While the aecXML initiative was aimed at producing a collection of transaction schemas, ifcXML did offer storage and transfer capabilities applicable to Collaborative systems. The Leeds University (2002) DocLink specification then extended the ifcXML model and applied this to the transfer of documents and associated metadata between collaborative systems, developing a series of data transfer transactions that could be executed in near real-time.

In contrast to previous work the generic model of a collaborative system designed by the NCCTP members, takes the existing structure of their repositories as a foundation, allowing vendors to more easily conform as no restructuring of their data is required. The Information defined in the NCCTP standard is divided into two main subtypes, the actual documents and structures which exist in the project, and the people who interact with these objects. Within a compliant XML document all definitions of users and the structures in which they are held, organisations and groups, are defined within the two NCCTP elements organisations and groups. Similarly all the information regarding documents is contained within the NCCTP element folders. A Diagram showing the main classes and their relationships of the NCCTP Standard can be seen in FIG. 5.

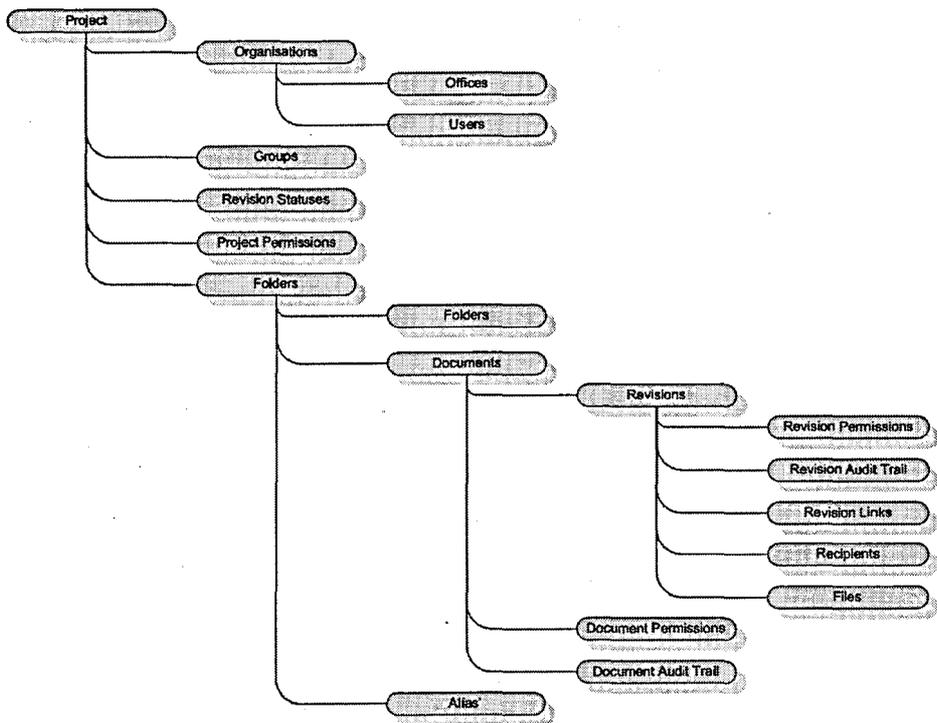


FIG. 5: Main Classes of NCCTP Standard

#### 4.1 IMPLEMENTATION OF THE BULK EXCHANGE STANDARD

To ensure that the proposed standard was suitable to allow project data to be exchanged between any collaborative systems it was important that vendors implemented it against their own systems as early as possible in the project. The NCCTP used a series of implementation workshops to improve the standard and ensure that it meets the needs of all vendors. Between each workshop individual vendors would attempt to implement that latest revision of the standard then bring any issues they encountered for discussion and rework at the next workshop.

During the implementation stage a number of test projects were created by each vendor, for intra-organisational testing of the project data exchange solution. Through the transfer of these test projects a series of key steps have been identified which will form the basis of any transfer of project data between collaborative systems. This common migration process is illustrated in FIG. 6, with each step explained in the remainder of this section.

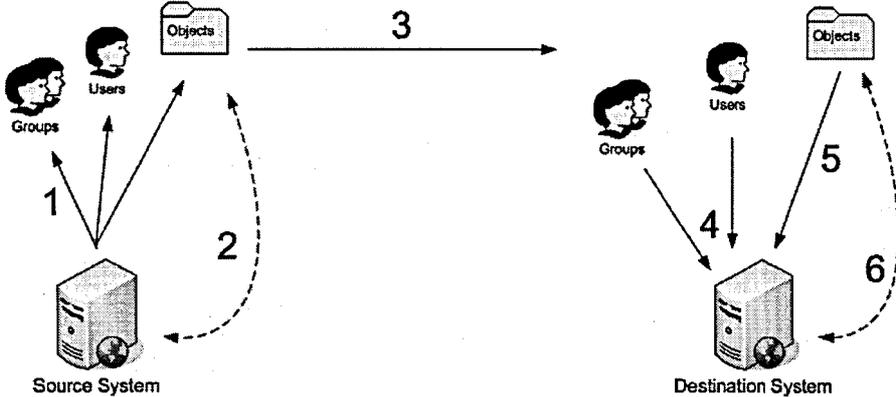


FIG. 6: Steps in the Bulk Transfer of Project Data

1) The selected project is extracted from source collaborative system and stored in the NCCTP XML format. This extraction includes all objects, files and versions along which the project participants and the groups are structured into.

2) A process is undertaken to verify that the extracted data is a true representation of the project as it existed in the source system. The level of verification done will depend upon the requirements of the client and the speed at which the transfer of data must be accomplished. During intra-organisational testing of the standard, vendors were required to complete detailed checklists to verify the data.

3) The method selected for the data transfer from source to destination system will depend on the quantity of project data and the speed the project has to be live on the destination system.

4) Importing/mapping of user and groups from the source collaborative system into the destination system. User or groups which already exist in the destination system will have to be mapped, while users and groups which do not exist will have to be created. When mapping users between source and destinations system care needs to be taken to ensure mappings are correct and that the same user could have different details on the two systems.

5) Objects, including folders, documents and revisions are imported into the destination collaborations system. User and group mapping information obtained from step 4 are used to recreate object access control lists and audit history.

6) A process is undertaken to validate that the information which has been imported into the system is a true representation of the project which was defined in the xml

document(s). The level of verification done on the imported data before end users are allowed to interact with it once again will depend upon how quick the client wishes to get the project data accessible to users again. During intra-organisational testing of the standard, vendors were required to complete detailed checklists to verify the data.

#### **4.2 PRACTICAL BARRIERS TO PROJECT TRANSFER BETWEEN DIFFERENT COLLABORATIVE SYSTEMS**

Even with standards in place a number of issues remain that place pressures on organisations to remain with the provider that they currently have. Some of these issues can be addressed by providers, by increasing the confidence in transfer mechanism. Several issues are a direct result of the differences that exist between systems and therefore cannot be resolved since each system has been developed over years and are unlikely to change the fundamentals of how the system works

As the number of project participants increase the task of ensuring correct user, group and organisation mapping becomes an enormous task. Although automated utilities could be written, the risk of getting it wrong and exposing documents to the wrong users would not be acceptable.

Different collaborative systems store object access control lists at different levels, which could prove problematic if moving from a more to less detailed structure. FIG. 7, shows an example of the potential data corruption which could occur when moving project data between systems, which support ACL's (Access Control List) at the document level and revision level respectively. The figure shows that when moving data to a more fine grained system scenario (1) no issues occur. However if data were transferred from a revision level permission system to the document level system scenario (2), the potential exists to expose content to the wrong users, or remove access rights to objects which existed in the source system.

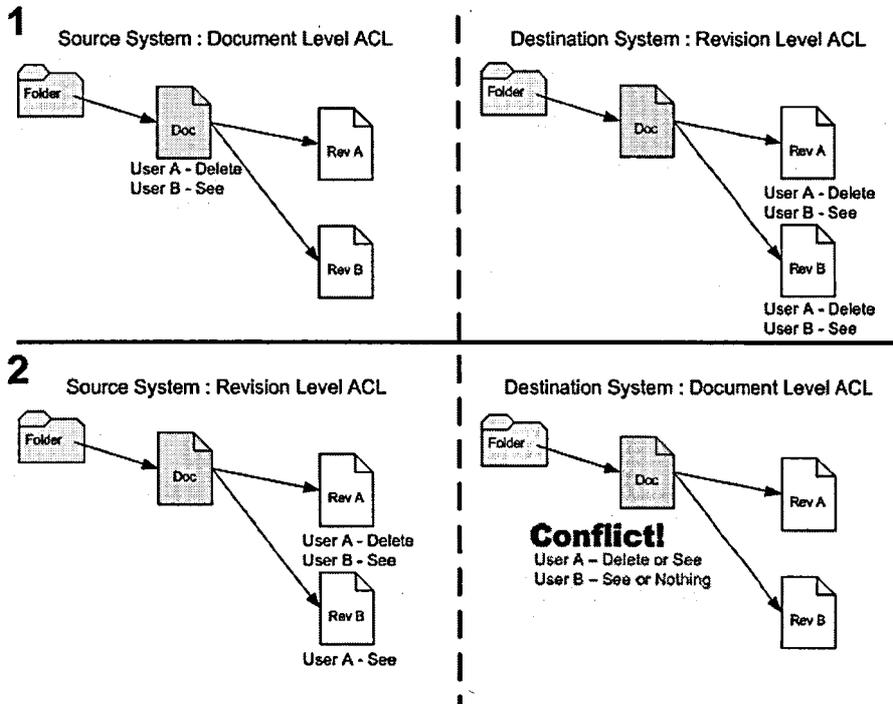


FIG. 7: Potential Data Corruption Issue when Migrating Access Control Lists

When transferring a large project between collaborative systems the amount of downtime required extracting, transferring and then re-importing the data may be too large to be acceptable to the client and their partners. Additionally as each different collaborative system has its own uniqueness, project data will never appear exactly as it did in the old system discouraging clients from transferring data in anything other than a real emergency. With the change of user interface and the loss of productive work that this could bring to the organisation, users not knowing how to use the interface correctly, the perception amongst users could be created that data has been lost during the transfer process.

## 5. FUTURE WORK

The work presented in this paper in defining the generic collaborative system used by the construction industry, can form the basis of a number of different future research paths. This is especially possible since the standard devised was the collective work of the majority of the key suppliers of this technology to the construction sector. The following sections outline the three most likely future projects which would build on this work.

### 5.1 EXTENDING THE STANDARD

The current version of the standard covers the core components, common to all vendors which make up a collaborative system. This work could be extended to define

some of the optional functionalities offered by some of the vendors, making it easier for other vendors to recognise the features in the XML output. Currently these extensions can be exported by the vendor but the schema does not define how they should be stored in XML.

## **5.2 INCREMENTAL PROJECT UPDATES**

The incremental replication of project data across a number of different project collaboration systems is a logical extension of the initial work done on bulk exchange. This enhancement would allow for users to access, although not in real-time, project data through a familiar interface. This is a logical extension since the objects being updated on each system are those already defined by the NCCTP standard, documents, revisions, access control etc.

## **5.3 REAL-TIME INTEGRATION**

Real-time integration of collaborative system offers users the ability to view, modify or delete information stored in a number of different collaborative systems from a single user interface. Unlike incremental project updates, outlined in section 5.2, there is no requirement to duplicate the information on all linked systems, and none of the delays are associated with the staged updating of information. Through its common definitions the NCCTP standard will help move towards this by offering vendors a generic way of describing the content which exists in their repository.

## **6. CONCLUSIONS**

With the development and implementation of the NCCTP bulk project exchange standard clients of collaborative solutions can now have the confidence that the ability exists to migrate live projects should the need arise. This reduces the risks associated with using these tools from the client's perspective, and increasing confidence in the project collaboration solution providers. This standard will allow for the common definitions now agreed upon by the NCCTP members to form the basis for greater level of interoperability in the future, allowing for the incremental exchange of data between the collaborative systems, and eventually the potential of real time access to information stored in different collaborative repositories from a single familiar system.

The ability to transfer project data between systems offers great benefits in post project analysis for those organisations which utilise an enterprise wide system. It allows for all users throughout the organisation, not only those who participated on the project, access to the data through their usual application.

While the developed standard in theory is capable of transferring projects at any time during their life cycle, it may only be practical to move this data in particular circumstances, due to the associated down time required for the migration. Likely opportunities for the bulk transferring project data would include:

- The vendor storing the projects is no longer able to offer the service so a new provider is required.
- A phase of the project is completed, and there is gap before the next phase starts.

- Project data is being transferred from an internal system to an external system to allow for collaborative working with other organisations.

Through the inter-organisational testing which has been conducted as part of the implantation of the exchange standard it is recommended that wherever possible the destination systems selected by clients should be structured as close as possible to the source system. The advantage being that it will allow users of the system to interact with the migrated objects more quickly than if this has been manipulated to fit into the new system.

## 7. ACKNOWLEDGEMENTS

The author wishes to thank the NCCTP organisation generally for facilitating the work presented in this paper and the following providers of collaborative technology who participated in the development of the NCCTP project data exchange standard, 4Projects, Aconex, Asite, BIW Technologies, Build Online, Business Collaborator, Cadweb, Causeway Technologies and Sarcophagus.

## 8. REFERENCES

- 4Projects & Tesco Express (2003), 4Project Press Release, [www.4projects.com](http://www.4projects.com), <http://www.4projects.com/Page.aspx?CAT=206>, (date accessed 29 November 2007)
- Aconex (2005), Aconex Joins the NCCTP, [www.aconex.com](http://www.aconex.com), <http://www.aconex.com/Corporate/News/Latest-News/Aconex-Joins-The-NCCTP.html>, (date accessed 28th November 2007)
- AutoDesk (2004), Autodesk Introduces Buzzsaw Server Edition, press release, 11 February 2004
- BCL (2004), BCL is a proud member of the newly formed NCCTP group, [www.groupbc.com](http://www.groupbc.com), <http://www.groupbc.com/news/items/ncctp0204.html>, (date accessed 28th November 2007)
- BCL (2007), Enterprise Collaboration, [www.groupbc.com](http://www.groupbc.com), <http://www.groupbc.com/product/enterprise.html>, (date accessed 30<sup>th</sup> November 2007)
- Berning and Flanagan (2003), E-Commerce and the Construction Industry: User Viewpoints, New Concerns, Legal Updates on Project Web Sites, Online Bidding and Web-Based Purchasing, [www.constructionweblinks.com](http://www.constructionweblinks.com), 22 December 2003, [http://www.constructionweblinks.com/Resources/Industry\\_Reports\\_Newsletters/Dec\\_22\\_2003/e\\_commerce.htm](http://www.constructionweblinks.com/Resources/Industry_Reports_Newsletters/Dec_22_2003/e_commerce.htm), (accessed 28 November 2007)
- Birkby G. and Nugent J. (2002), The ASP with a String in its Tail, Building, 8 June, pp50-51
- Björk B-C., Huovila P., Hult S. (1993) Integrated Construction Project Document Management (ICPDM), Proceedings of EuroPIA '93, Delft, June 21-24, pp 135-146.
- Building (2004) Revenue Probes IT Provider BIW, Building Magazine Issue 37, 24<sup>th</sup> September 2004, p 14.
- Cutler (2006) Information Aggregation, AEC Magazine January/February 2006, pp 20-21,
- Egan J (1998) Rethinking construction: The report of the construction task force, Department of the Environment, Transport and the Regions, London. p16.
- Hajjar D. & AbouRizk S.M. (2000) Integrating Document Management with Project and Company Data, ASCE Journal of Computing in Civil Engineering, January, 14(1), pp 70-77.
- Hampton J. (2001) Stung, Construction Manager, September, pp. 8-10

- IAI (2001) ifcXML recommended standard, [http://www.iai-international.org/iai\\_international/Technical\\_Documents/IFCXML.htm](http://www.iai-international.org/iai_international/Technical_Documents/IFCXML.htm)
- IAI (2004) aecXML standard documents, <http://www.iai-na.org/aecxml/documents.php>
- ISO (2000a) Management Data (Metadata) for Technical Documents, IEC/ISO 82045, 2000
- ISO (2000b) Technical Product Documentation – Metadata for Construction Documentation, PD ISO 19033, 2000.
- Krajewski K. (2001) What's up with ASPs?, Association Management, Volume 53, 2001, p. 38-46.
- Lane B. (2003), The Collaborators, Building Magazine Issue 28, 18<sup>th</sup> July 2003, p. 53
- Leeds University (2002) The DocLink Specification, <http://www.doclink.info/>
- McBride K. (2003) 'Clicks and Bricks: Practical Technology in Construction Contracts', Plan Magazine, October.
- NCCTP (2004) The NCCTP Project Exchange Standard, <http://www.ncctp.net/>
- NCCTP (2005), Asite joins the NCCTP, Constructing Excellence, <http://ncctp.constructingexcellence.org.uk/article.jsp?id=8976>, (date accessed 28 November 2007)
- NCCTP (2006), Proving Collaboration Pays Study Report, Constructing Excellence, June 2006. pp 31
- Rezgui Y., Cooper G. (1998) A Proposed open infrastructure for construction project document sharing, Electronic Journal of Information Technology in Construction (ITcon). 3. pp 11-24.
- Wilkinson P. (2005), Construction Collaboration Technologies – The Extranet Evolution, Taylor & Francis, pp 8
- Yeomans S., Bouchlaghem D. and El-Hamalawi A., (2005), Collaborative Extranet working via multiple construction project extranets, Proceedings of the 3<sup>rd</sup> International Conference on Innovation in Architecture, Engineering and Construction, 15 – 17<sup>th</sup> June 2005, pp 218

## **APPENDIX E NCCTP BULK DATA EXCHANGE STANDRAD**

---

# NCCTP Bulk Project Data Exchange Specification

---

NCCTP Specification Request 1  
version 1.0  
20 August 2005

Acknowledgements.....	155
License.....	156
<b>1 Preface.....</b>	<b>158</b>
1.1 Documents Included.....	158
<b>2 Introduction.....</b>	<b>160</b>
2.1 Motivation.....	160
2.2 Goals.....	160
<b>3 Use Cases.....</b>	<b>162</b>
3.1 Consolidation of Data into a Central Repository.....	162
3.2 Emergency Project Reactivation.....	162
<b>4 The Generic System model.....</b>	<b>165</b>
4.1 Project Participants.....	165
4.1.1 Organisations.....	165
4.1.1.1 Offices.....	165
4.1.1.1.1 Addresses.....	166
4.1.2 Users.....	166
4.1.2.1 Addresses.....	166
4.1.3 Groups.....	166
4.2 Object Nodes.....	167
4.2.1 Projects.....	167
4.2.1.1 Addresses.....	167
4.2.2 Folders.....	167
4.2.3 Documents.....	168
4.2.3.1 Revisions.....	168
4.2.3.1.1 Files.....	168
4.2.4 Alias'.....	168
4.3 Access Control Lists.....	168
4.3.1 Permission Model.....	169
4.3.2 Cascading of Permissions.....	169
4.3.3 NCCTP defined Permissions.....	170
4.4 Audit History.....	170
4.4.1 NCCTP Defined Audit Events.....	170
4.5 Revision Linking.....	171
4.5.1 Revision Recipients.....	171
4.5.2 Revision Links.....	171

## ACKNOWLEDGEMENTS

This specification is the collaborative product of

Scott Moses (specification lead, Causeway),

Scott Moses (author, Causeway)

And other members of the NCCTP expert Group, as well as many others who contributed with corrections and suggestions.

## **LICENSE**

This specification may only currently be used by ncctp member organisations.

---

## 1 PREFACE

---

This is Version 1.102 of the NCCTP Data Exchange Standard.

### 1.1 DOCUMENTS INCLUDED

---

The specification Includes:

- This document in Adobe Portable Document Format (ncctpDXS-1.102.pdf).
- The ncctp data exchange XML schema document (NCCTP\_v1\_102.xsd).
- The classes and interrelationships document (NCCTP Standard Document.doc)
- Provider Abbreviations Document

In case of a discrepancy between this document and the NCCTP Data Exchange Schema, then the NCCTP Data Exchange Schema document should be considered normative.

---

## 2 INTRODUCTION

---

### 2.1 MOTIVATION

---

The NCCTP (Network for Construction Collaboration Technology Providers) formed in December 2003. The current members of the network, managed by CIRIA, are:

4Projects, Asite, BIW Technologies, BuildOnline, Business Collaborator, Cadweb, Causeway Technologies and Sarcophagus.

The network aims to promote the effective use of online technology to support collaborative working on projects and capital developments in UK construction. It aims to increase interoperability between systems — engendering easy transfer of data through definition and adoption of standards. The Network will promote the benefits of using collaborative technology and demonstrate the value of collaborative working. Importantly, it provides a single independent body with whom clients can communicate regarding the future development of collaboration technology.

### 2.2 GOALS

---

The NCCTP seeks to promote the benefits and use of collaborative technology in the construction and related industries, by achieving the following objectives:

- To develop and implement an agreed set of data exchange standards between all members to enable bulk transfer of data from any one system to another.
- To develop and implement an agreed set of data exchange standards between all members to enable routine transfer of information between systems for cross project working.
- To establish a group whose membership broadly represents the collaborative technology providers working within the construction industry and provides a vehicle to address generic market and technology issues.

---

### **3 USE CASES**

---

#### **3.1 CONSOLIDATION OF DATA INTO A CENTRAL REPOSITORY**

---

Quick Fix Joiners are a large national organisation supplying joinery services to the construction industry. They have an enterprise wide content management system, which they use to coordinate activities across the entire business. The companies' employees collaborate with many other organisations on projects throughout the country; however since they are not the main contractor or client different extranet systems are used on different projects. At the completion of their involvement with the project Quick Fix Joiners request a copy of the data that they had access to, so they can make it available to their entire organisation, leveraging the knowledge gained on the project.

Upon receipt of the request the company that initially set up the project extranet makes a request to the vendor for a copy of the project data, Since the vendor is a member of the NCCTP and fully supports the bulk data exchange standard they are able to provide the company with a copy of the project data, which they then forward to Quick Fix Joiners.

When the data is received Quick Fix Joiners pass this data on to their enterprise system provider, who also supports the bulk data exchange standard, to import into their enterprise system. Once the data has been integrated into the system project coordinators from Quick Fix Joiners give none project participants appropriate access rights to the data.

Thus the knowledge gained during the participation in the project is retained within the organisation.

#### **3.2 EMERGENCY PROJECT REACTIVATION**

---

A road network maintenance company successfully won a 10 year contract to maintain a 50 mile section of a major roadway, which over the past two years has been managed using a project extranet system. Recently however the extranet system provider has been experiencing difficulties maintaining their service level and is in danger of going out of business. The road maintenance company wishes to move it data to a more stable supplier and comes to an agreement to get a copy of the project data in NCCTP format

With the project no longer live of the existing extranet system, the company is eager to quickly reactive the project on a new vendors system. To minimise staff retraining they select a system to is as close as possible to the system that they are migrating from, and one that supports the data exchange standard.

The data is migrated and the project is now accessible to all the users from the previous system.



## **4 THE GENERIC SYSTEM MODEL**

---

### **4.1 PROJECT PARTICIPANTS**

---

This Section contains items that are directly associated with the participants of the project.

#### **4.1.1 ORGANISATIONS**

---

A Container that holds users who belong to the same organisation from the perspective of the collaborative system, allowing the user account to appear with greater context for other participants on the project. If an exporting collaborative system holds multiple projects which each derive their participant list from a commonly held set of organisations then special assessment of which organisations to include in the export will be needed. A valid extraction of organisation should only include those that have users who are currently, or have previously participated on the project, or are participants or have previously participated in the project collectively.

For example 'a steel maker' organisation is currently a participant of the 'tall building' project, therefore when exporting the project this organisation and all its users should be included in the XML Document outputted from the system. However the project also includes 'sample user' as a participant, who belongs to the 'building company' organisation which itself is not in the projects participants list, therefore on export the organisation should be included, but only contain the user account of 'sample user'.

Note: the organisationID given to any organisation included in the export XML Document must be unique amongst all other organisations exported within the project.

##### **4.1.1.1 OFFICES**

---

An Organisation's offices are the physical locations in which its users work. Each organisation may have any number of separate offices, including zero, but should only specify one of these offices as their main office. If user details include reference to an office then is office must be present within the same organisation as the user. For Example a user who is a child of one organisation cannot be associated with an office that is a child of another organisation.

Note: the officeID given to any office within the entire project exported must be unique amongst all offices; no two office elements can have the same officeID value.

#### 4.1.1.1 ADDRESSES

---

An organisation office is a physical location that will have an associated postal address, which the standard allows to be defined as a child of the office element. Exports may optionally include address details for the offices that are defined within the exported project data. This address is of the same type as those that can be associated with both the project and individual users.

#### 4.1.2 USERS

---

The User element holds details about the system account of a user who is or has been associated with the project, which are grouped by the organisation to which they belong. Users are the individual people that work with the data contained in the project, uploading, downloading and viewing documents. Users are referenced from many different objects in the NCCTP System, and references cannot be made to any users who do not exist in the export.

To allow for this rule about every user being present in the XML the concept of a Power User who is a users who is created by themselves.

Note: the userID given to any user must be unique amongst all users that exist within the exported project, regardless of which organisation they belong to.

#### 4.1.2.1 ADDRESSES

---

The postal location contact details of the user account stored in the system can be used when address information is specified on a user level. User address information can be included along with reference to an office with its own postal address information, or in place of an office reference. Indeed a user can exist with no personal postal address or link to office.

#### 4.1.3 GROUPS

---

Groups are a collection of users or other groups that exist within a project for the purpose of Access Control Lists or Document distribution. A Group can contain any number of children that can exist

within any of the organisation associated with the project. However Groups cannot contain Organisations as members, and if the exporting system allows this then the exporter must replace this group member accordingly. The following 2 solutions are recommended for the replacement of organisations in groups:

Solution 1: Whenever an organisation is a member of a group, simply add all members of that organisation to the group as individuals.

Solution 2: Create a group that contains all the members of the organisation and add this as a member of the group as opposed to the organisation.

## **4.2 OBJECT NODES**

---

This Section contains information relating any of the nodes that exist within the project.

### **4.2.1 PROJECTS**

---

Contains all the objects related to the collaborative project.

#### **4.2.1.1 ADDRESSES**

---

The Postal location contact details of the project that is being transferred, if the exporting system contains information about where the project is located then this should be store here. Each project can only have one address entry and any system that holds multiple addresses for any project should just export the main address to a compliant NCCTP XML Document.

### **4.2.2 FOLDERS**

---

Folders Objects define the structure in which documents are filed, and maybe nested infinitely within the project workspace. Folder objects can contain any number of child nodes supported by the ncctp schema, but must have a unique id property amongst all other folder objects contained within the project.

Vendor systems that do not contain the concept of folders will need to create temporary transfer folder objects to store all documents, with the recommendation that documents should be logically grouped into suitable folders.

### 4.2.3 DOCUMENTS

---

Document Objects are containers that hold one two many revisions of that document. Each Document Object contained in the exported project must have a unique id property amongst all other document objects.

#### 4.2.3.1 REVISIONS

---

Revisions of a document, these objects are containers that hold the file(s) associated with the revision. Each revision that exists within the exported project must have a unique id property amongst all other revision objects that exist within the project.

The Value of the revisionStatus property of the revision must be one of those revision statuses defined within the project exported, held within the revisionStatuses container.

#### 4.2.3.1.1 FILES

---

A file associated with a revision through its parent child relationship, it does not include the file data, only information about the file, and its location in an external file store. Each file object that exists within the exported project must have a unique id property amongst all other file objects that exist in the project.

#### 4.2.4 ALIAS'

---

Shortcuts to Documents which are stored in a different location within the project folder structure, allowing user's quick navigation to documents that are stored in disparate locations. Only aliases to document are supported in the current version of the schema, should the system allow for aliasing of other node types then these cannot be handled by the alias class.

Note: The aliasID assigned to an alias must be unique amongst the entire set of alias' that exist within the exported project.

Any aliases to objects that are stored outside of the project being extracted are not supported although exporters could add an additional folder to the export to hold these objects. However causation must me used to ensure that these objects do not reference any other objects or users that exist outside of the project.

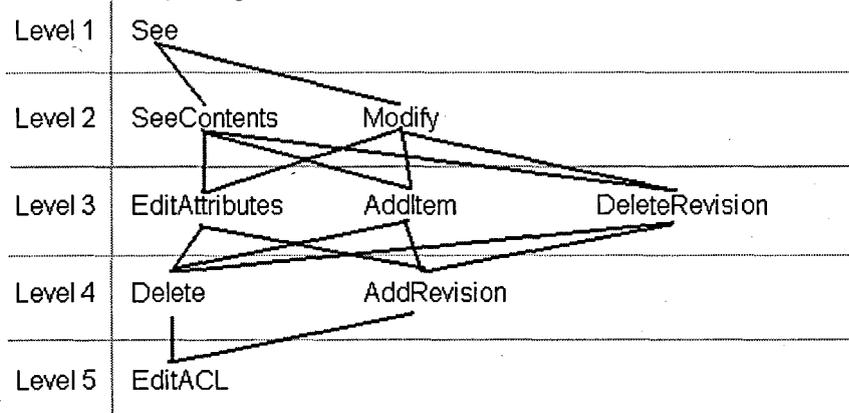
### 4.3 ACCESS CONTROL LISTS

---

User's rights to interact with the objects that are contained within the project are stored in access control lists. These objects can be the child of any of the following objects, project, folder, document, revision and file. The access control list contains one entry for each user permission profile that can be assigned to an object, which can be either a user, group or organisation reference.

### 4.3.1 PERMISSION MODEL

The model consists of a series of privileges that a specified user, group or organisation has on an object. The privileges are split into a number of levels which are logically built upon each other. Therefore for a have privileges in that exist in the third level they must have certain level one and level two privileges as well.



Example 1 – To have permissions to edit Access Control Lists that user must have permission to do all other actions on the object as well. This makes perfect sense because if a user could change the access control list then they could give themselves the other permissions anyway.

Example 2 – If a user has permissions to add Items to an object the must have permissions to See the object see the contents of the object and modify the object. This again makes sense as if the user was unable to see the object then they would not be able to find it to add the other object to it.

### 4.3.2 CASCADING OF PERMISSIONS

The permissions applied to an object shall be inherited by all children unless another access control list is applied to that child specifically. Through the application of a new access control list an object will not get any of the permissions that existed on the parent node only those that are defined in the new access control list. Therefore a new access

control list cannot be used to amend the permissions that exist on an object only define them in totality.

Example 1 – In a system that supports Access Control Lists at the document level, and not the revision level, it should be assumed that the Access Control List on the document is applied to all revisions, and files.

Example 2 – A Folder Object Contains 3 Documents, 2 of which have an Access Control List defined; only the document that does not have an access control list should inherit the list from the folder.

### **4.3.3 NCCTP DEFINED PERMISSIONS**

---

The following is a list of permission that can be granted to users, groups or organisation on objects in the system:

see – See the name of the item listed in a browser page.

seeContents – See contents of the item (i.e. see a list of items within a container or view/fetch/download a document).

modify – Modify the name, description or configuration of an object.

editACL – Open the permissions page for that object and set access control permissions for other groups and users (This provides total access to the item).

editAttributes –

addItem – Add Objects to this object available only for projects and folders.

deleteRevision – Allows user to delete a revision of an item.

Delete – Delete the Item.

addRevision – Reserve the item allowing the user to create new revisions.

### **4.4 AUDIT HISTORY**

---

An Objects Audit history records actions that have taken place on the object since it was created, the time that these actions occurred and the performer of the action. The first chronological action on any object should be the create action, and no other action should happen before this. All users referenced in the audit history must be contained in the exported project, even if they have since been deleted or removed from the project.

#### **4.4.1 NCCTP DEFINED AUDIT EVENTS**

---

The following is a list of allowable audit events defined by the NCCTP Standard:

Upload – An object is uploaded to the system.

Download – An object is downloaded from the system.

Revise – An object is revised in the system.

Modify – An objects properties are changed in the system.

Delete – An object is deleted from the system.

Create – An object is created in the system.

View – An object in the system is viewed.

Reserve – An object in the system is reserved by a user.

Unreserve – An object in the system is unreserved by a user.

## **4.5 REVISION LINKING**

---

Revision Linking provides support the linking individual revisions with other revisions or project participants.

### **4.5.1 REVISION RECIPIENTS**

---

This is a distribution mechanism that allows for access to be granted a particular revision for a number of project participants. All information that is contained within these links should be replicated in the Access Control List for the revision in question. All references to project participants in these links should be contained in the export project document.

### **4.5.2 REVISION LINKS**

---

Revision Links enables association/attachments/XREFs between revisions, allowing each revision to link to many others. Revision Links is an optional child of revision but if included must reference at least one other revision. All revisions reference in revision links must exist with the project that has been export.

**NCCTP STANDARD**  
**CLASSES AND INTERRELATIONSHIPS DOCUMENT**

**V3.1**

**28TH MARCH 2006**

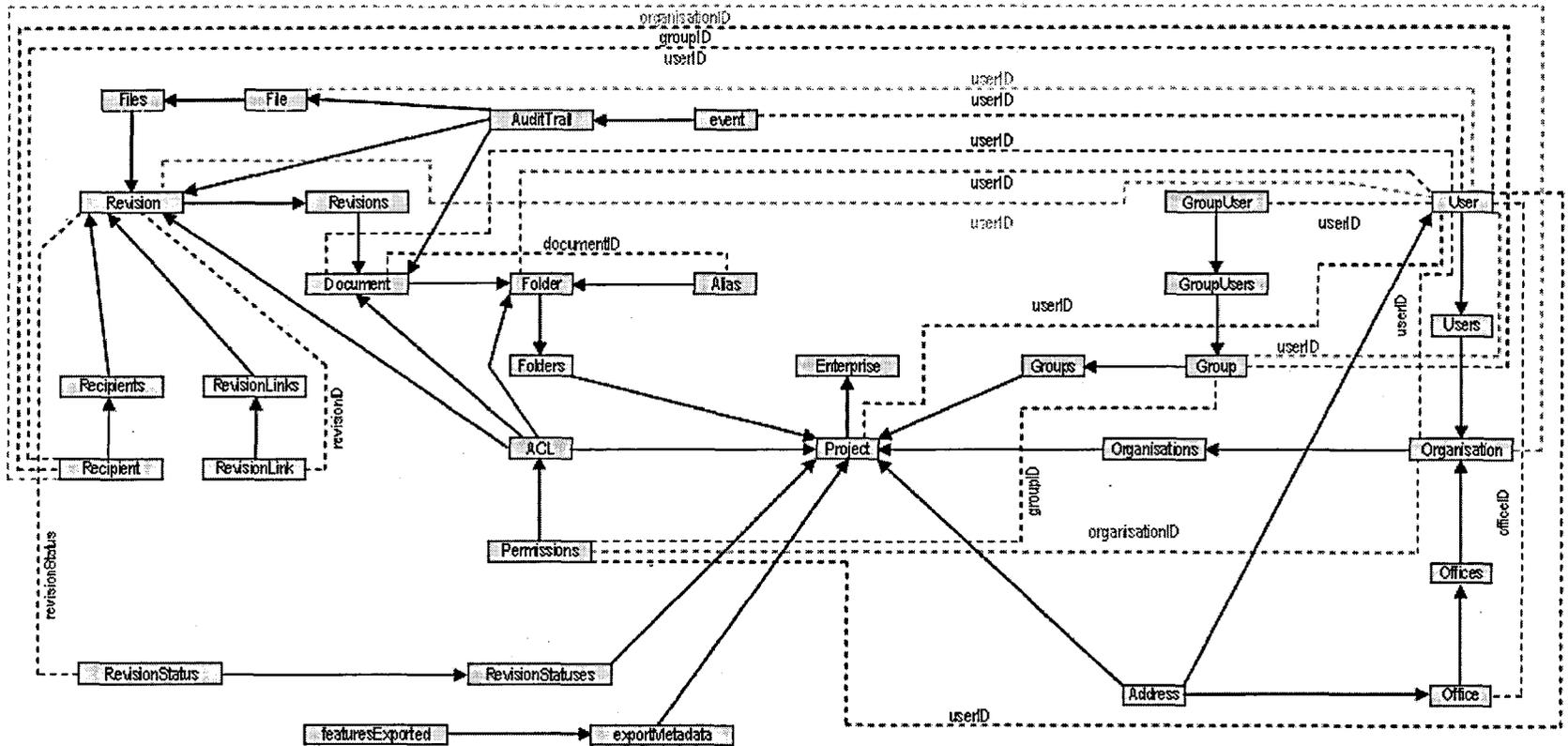
**CREATED BY: SCOTT MOSES**

**EDITED BY: SCOTT MOSES**

This document details the object classes and their relationships that make up the NCCTP Bulk Data Exchange Standard.

VERSION	AUTHOR	ORGANISATION	DATES	NOTES	DISTRIBUTION LIST
1.0	Scott Moses	Causeway	6 <sup>th</sup> January 2005	Draft Documentation	Distributed via the NCCTP web group
2.0	Scott Moses	Causeway	4 <sup>th</sup> April 2005	Additional Column added to tables	Distributed via the NCCTP web group
3.0	Scott Moses	Causeway	27 <sup>th</sup> September 2005	Changes for v1.102 of the data exchange standard	Distributed via the NCCTP web group
3.1	Scott Moses	Causeway	28 <sup>th</sup> March 2006	Front Page and Headers added to the document	Distributed via the NCCTP web Group

## Classes and Interrelationships Diagram



Element Name	Element Description	Type	Required	Min Occurs	Max Occurs	Supported
--------------	---------------------	------	----------	------------	------------	-----------

<b>Enterprise</b>			M	1	1	
Root Element of the schema contains all other elements						
project	The project holds all information regarding the users, group, organisations, folder and documents that are associated with the project		M	1	1	

<b>project</b>						
Holds all of the information that relating to the project						
projectID	System ID of the project	PK xs:string	M	1	1	
exportMetadata	Container that holds information about the project that has been exported		M	1	1	
name	Project Name for display purposes, should be unique to each client	xs:string	M	1	1	
startDate	The calendar date which the project started	xs:date	O	0	1	
endDate	The calendar date which the project is due to end	xs:date	O	0	1	
projectNotes	Any miscellaneous project notes	xs:string	O	0	1	
status	Indicates the current status of the project can either be 'active' or 'deleted'	ref to status	M	1	1	
clientID	Client that the project belongs to	xs:string	M	1	1	
ownerID	Link to user who owns the project	FK xs:string	O	0	1	
clientProjectID	The clients internal project ID, differs from the projected as this is system generated	xs:string	O	0	1	
createdByID	Link to user who created the project on the system	FK xs:string	O	0	1	
dateCreated	The date and time that the project was created on the system	xs:dateTime	O	0	1	
lastModifiedByID	Link to the last user who modified the project on the system	FK xs:string	O	0	1	
dateLastModified	The date and time that the project was last modified on the system	xs:dateTime	O	0	1	
##Other	Allows additional project metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*	
organisations	Container that holds all of the organisations that are associated with the project		M	1	1	
groups	Container that holds all of the groups that are defined by the project		M	1	1	
address	Details about the physical location of the project		O	0	1	
revisionStatuses	Container that holds a list of statuses that that can be attached to revisions		O	0	1	
acl	Contains information about the rights of users, groups and organisations on the project. Until other acs are defined this information is cascaded down through the folder structure		M	1	1	
folders	Folders define the structure for the filling of documents. For very large xml files the document elements can be defined within external xml files These files are referenced in the main document by the documentRef element The string contained within the element will point to the location of the of the external xml file that defines the document		M	1	1	

<b>exportMetadata</b>						
Information about the exported data						
featuresExported	Container that holds information about the objects contained in the project		M	1	1	
dateOfExport	When the project was exported from the source system	xs:date	M	1	1	
exportNotes	Information about the project exported that could be of use to the destination system.	xs:string	M	1	1	
sourceSystem	Tag that identifies the source of the project data	xs:string	M	1	1	

## Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

<b>featuresExported</b>						
Indicates the features that are supported in this export						
project	true if this class of object is included in the export	xs:boolean	M	1	1	
organisations	true if this class of object is included in the export	xs:boolean	M	1	1	
users	true if this class of object is included in the export	xs:boolean	M	1	1	
offices	true if this class of object is included in the export	xs:boolean	M	1	1	
projectAddress	true if this class of object is included in the export	xs:boolean	M	1	1	
userAddress	true if this class of object is included in the export	xs:boolean	M	1	1	
officeAddress	true if this class of object is included in the export	xs:boolean	M	1	1	
groups	true if this class of object is included in the export	xs:boolean	M	1	1	
revisionStatuses	true if this class of object is included in the export	xs:boolean	M	1	1	
projectACL	true if this class of object is included in the export	xs:boolean	M	1	1	
folderACL	true if this class of object is included in the export	xs:boolean	M	1	1	
documentACL	true if this class of object is included in the export	xs:boolean	M	1	1	
revisionACL	true if this class of object is included in the export	xs:boolean	M	1	1	
documentAuditTrail	true if this class of object is included in the export	xs:boolean	M	1	1	
revisionAuditTrail	true if this class of object is included in the export	xs:boolean	M	1	1	
fileAuditTrail	true if this class of object is included in the export	xs:boolean	M	1	1	
revisionLinks	true if this class of object is included in the export	xs:boolean	M	1	1	
revisionRecipients	true if this class of object is included in the export	xs:boolean	M	1	1	
folders	true if this class of object is included in the export	xs:boolean	M	1	1	
documents	true if this class of object is included in the export	xs:boolean	M	1	1	
aliases	true if this class of object is included in the export	xs:boolean	M	1	1	
revisions	true if this class of object is included in the export	xs:boolean	M	1	1	
files	true if this class of object is included in the export	xs:boolean	M	1	1	
multipleFiles	true if this class of object is included in the export	xs:boolean	M	1	1	

<b>organisations</b>						
A container that holds all of the organisations that are associated with the project						
organisation	Details of a single organisation		M	1	*	

<b>organisation</b>						
Details of a single organisation that is associated with the project						
organisationID	System ID of the organisation, must be unique amongst all organisations	PK xs:string	M	1	1	
name	The name that is stored within the organisation should be unique.	xs:string	M	1	1	
primaryContact	Organisations primary contact stored within the system	xs:string	O	0	1	
website	Main Organisational website	xs:anyURI	O	0	1	
status	Indicates the current status of the organisation either 'active' or 'deleted'	ref to status	M	1	1	
createdByID	Link to user who created the organisation on the system	FK xs:string	O	0	1	
dateCreated	The date and time that the organisation was created on the system	xs:dateTime	O	0	1	
lastModifiedByID	Link to the last user who modified the organisation on the system	FK xs:string	O	0	1	
dateLastModified	The date and time that the organisation was last modified on the system	xs:dateTime	O	0	1	
##Other	Allows additional organisation metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*	
users	A container that holds details of all the users that are associated with this organisation		M	1	1	
offices	A container that holds details of offices that are associated with the organisation		M	1	1	

<b>users</b>						
A Container that holds all the users who are members of a particular organisation						
user	information about an individual user who is a participant in the project		M	1	*	

<b>user</b>						
users account and contact information						
userID	System ID of the user, must be unique amongst all users exported	PK xs:string	M	1	1	
loginName	The name used by the user to gain access to the system, sign-in name.	xs:string	M	1	1	
firstName	The users first name	xs:string	M	1	1	
middleName	The users middle name	xs:string	O	0	1	
lastName	The users last or family name	xs:string	M	1	1	
email	The users primary email address	xs:string	M	1	1	
phoneNumber	The users phone number	xs:string	O	0	1	
faxNumber	The users fax number	xs:string	O	0	1	
mobileNumber	The users mobile number	xs:string	O	0	1	
officeID	Link to an office that is owned by the organisation	FK xs:string	O	0	1	
jobTitleRole	The users job title or role	xs:string	O	0	1	
status	Indicates the current status of the user either 'active' or 'deleted'	ref to status	M	1	1	
createdByID	Link to user who created the user on the system	FK xs:string	O	0	1	
dateCreated	The date and time that the user was created on the system	xs:dateTime	O	0	1	
lastModifiedByID	Link to the last user who modified the user on the system	FK xs:string	O	0	1	
dateLastModified	The date and time that the user was last modified on the system	xs:dateTime	O	0	1	
lastLogin	Date and Time that the user last logged into the system	xs:dateTime	O	0	1	
lastPasswordChange	Date and Time when the users password was last changed	xs:dateTime	O	0	1	
timeZone	The time zone that the user is located in	xs:string	O	0	1	
language	The language used by the user	xs:string	O	0	1	
##Other	Allows additional user metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*	
address	Address information held for the user on the system		O	0	1	

<b>address</b>						
Physical location of office, project or user. Country : should be entered in accordance with ISO 3166-1						
address1	Line 1 of the address	xs:string	M	1	1	
address2	Line 2 of the address	xs:string	O	0	1	
address3	Line 3 of the address	xs:string	O	0	1	
town	Town	xs:string	M	1	1	
state-Country	The State or Country in which the location is	xs:string	O	0	1	
country	The country in which the location is	xs:string	O	0	1	
zip-postCode	The Zip Code or Post Code of the location	xs:string	O	0	1	

<b>offices</b>						
A container that holds a number of offices that are associated with an organisation						
office	Details of an individual office		O	0	*	

<b>office</b>						
Details of a single office that is associated with an organisation on the project						
officeID	System ID of the office, must be unique amongst all offices	PK xs:string	M	1	1	
description	description of the office	xs:string	O	0	1	
isMainOffice	Indicates weather this office is the organisations main office, 'true' or 'false'.	xs:boolean	M	1	1	
phoneNumber	The main phone number of the office	xs:string	O	0	1	
faxNumber	The main fax number of the office	xs:string	O	0	1	
##Other	Allows additional user metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*	
address	postal address details for the office		M	1	1	

<b>groups</b>						
A Container that holds the groups that are associated with the project (project groups)						
group	Individual Group defined within the project		O	0	*	

## Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

<b>group</b>					
Details of a single group that is associated with the project					
groupID	System ID of the group, must be unique amongst all groups	PK xs:string	M	1	1
name	The display name assigned to the group, should be unique amongst all project groups	xs:string	M	1	1
status	Indicates the current status of the group either 'active' or 'deleted'	ref to status	M	1	1
createdByID	Link to user who created the group on the system	FK xs:string	O	0	1
dateCreated	The date and time that the group was created on the system	xs:dateTime	O	0	1
lastModifiedByID	Link to the last user who modified the group on the system	FK xs:string	O	0	1
dateLastModified	The date and time that the group was last modified on the system	xs:dateTime	O	0	1
##Other	Allows additional group metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*
groupUsers	A container that holds details of all the users who are members of the group		M	1	1

<b>groupUsers</b>					
A container that holds the members of a group					
groupUser	Reference to an individual member of the group		O	0	*

<b>groupUser</b>					
Details of a single member that is a member of a group					
groupUserID	System ID that identifies the user as part of the group	PK xs:string	M	1	1
userID or groupID	Reference to the ID of a user specified within the user element. The userID or groupID must be that of a user or group that is contained within the XML export document.	FK xs:string	M	1	1
status	Indicates the current status of the user within this group either 'active' or 'deleted'	ref to status	M	1	1

<b>revisionStatuses</b>					
A container that holds the allowable status for which revisions may be issued					
revisionStatus	Individual status that a revision may be issued for		M	1	*

<b>revisionStatus</b>					
Details a status for which revisions can be issued within this project					
statusID	System ID of the revision status, must be unique amongst all revision statuses	PK xs:string	M	1	1
description	A string that highlights the revision statuses display name	xs:string	M	1	1
status	Indicates the current status of the revision status 'active' or 'deleted'	ref to status	M	1	1

<b>acl</b>					
Access Control List for an object (Project, Folder, Document, etc.). A container that can hold one-to-many permissions detailing rights on the parent object.					
permissions	Holds the rights for an individual actor (Organisation/Group/User) on a object.		M	1	*

<b>permissions</b>						
Details the rights of an individual actor on an object within the system						
originatorID	System ID of the user who granted the permission	PK xs:string	M	1	1	
groupID or userID or organisationID	One and only one of these to be included referencing the system ID of a group, user or organisation. Indicates to what the permission applies to and must match to a group, user or organisation that is defined within the XML document.	FK xs:string	M	1	1	
see	See the name of the item listed in a browser page.	xs:boolean	M	1	1	
seeContents	See contents of the item (i.e. see a list of the items within the container, or view/fetch/download a document).	xs:boolean	M	1	1	
modify	Modify the name, description or configuration (catalog, list item or hidden) of the item.	xs:boolean	M	1	1	
editACL	Open the permissions page and set access control permissions for other users and groups (This provides total access to the item!)	xs:boolean	M	1	1	
editAttributes	Allows users to adjust the attributes of an object (Change the description associated with it).	xs:boolean	M	1	1	
addItem	Add items to this item (available only for folder and compound documents).	xs:boolean	M	1	1	
deleteRevision	Delete revision of the item.	xs:boolean	M	1	1	
delete	Delete the item.	xs:boolean	M	1	1	
addRevision	Reserve the item, allowing you to create new revisions.	xs:boolean	M	1	1	

<b>folders</b>						
A container that holds a number of folder elements. (Zero to Many folder elements)						
folder	Folders define the structure for the filing of documents.		O	0	*	

<b>folder</b>						
A container for folders, documents and alias' that defines the structure in which documents are stored						
folderID	System ID of the folder.	PK xs:string	M	1	1	
name	folders display name	xs:string	M	1	1	
description	description of the folder	xs:string	O	0	1	
sequence	Allows for the custom ordering of folders displayed to the user	xs:string	O	0	1	
fullQualifiedName	As an aid when using hierarchal folders	xs:string	O	0	1	
status	Indicates the current status of the folder either 'active' or 'deleted'	ref to status	M	1	1	
createdByID	Link to user who created the folder on the system	FK xs:string	O	0	1	
dateCreated	The date and time that the folder was created on the system	xs:dateTime	O	0	1	
lastModifiedByID	Link to the last user who modified the folder on the system	FK xs:string	O	0	1	
dateLastModified	The date and time that the folder was last modified on the system	xs:dateTime	O	0	1	
##Other	Allows additional folder metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*	
acl	Access Control List for the folder.		O	0	1	
auditTrail	Container that holds all actions that have been performed of this document.		O	0	1	
folder	New folder definition, allows nesting of folders to form the hierarchical folder structure		O	0	*	
document	Details of a document within this folder.		O	0	*	
documentRef	Reference to a document that is defined within another XML document, in order to reduce the size of the main XML document.	xs:string	O	0	*	
alias	Shortcut to a document that exists within another folder.		O	0	*	

## Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

<b>document</b>						
Contains one-to-many revisions						
documentID	System ID of the document, must be unique amongst all documents	PK xs:string	M	1	1	
name	Documents display name as seen by users, should be unique within its parent folder.	xs:string	M	1	1	
description	description of the document held within the system	xs:string	O	0	1	
status	Indicates the current status of the document either 'active' or 'deleted'	ref to status	M	1	1	
ownerID	Link to user who owns the document.	FK xs:string	O	0	1	
createdByID	Link to user who created the document on the system	FK xs:string	M	1	1	
dateCreated	The date and time that the document was created on the system	xs:dateTime	M	1	1	
lastModifiedByID	Link to the last user who modified the document on the system	FK xs:string	O	0	1	
dateLastModified	The date and time that the document was last modified on the system	xs:dateTime	O	0	1	
##Other	Allows additional document metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*	
acl	Access Control List for the document.		O	0	1	
auditTrail	Container that holds all actions that have been preformed of this document.		O	0	1	
revisions	Container that holds all of the revisions of this document.		M	1	1	

<b>auditTrail</b>						
A container that holds all of the events that have happened to an object (document, revision, file).						
event	Individual action that that happened to the object.		M	1	*	

<b>event</b>						
Details of a single event that has been recorded on the associated object						
eventID	System ID of the event.	PK xs:string	M	1	1	
action	Name of the action that was undertaken on the object. Must be one of the following 'Upload', 'Download', 'Revise', 'Modify', 'Delete', 'Create', 'View', 'Reserve' and 'Unreserve'.	xs:string	M	1	1	
userID	Link to the user who preformed the action on the object.	FK xs:string	M	1	1	
notes	description of the event	xs:string	O	0	1	
dateStamp	time and date that the event happened	xs:dateTime	M	1	1	

<b>revisions</b>						
Container that holds the revisions of a particular document.						
revision	Details an individual revision of the document.		M	1	*	

<b>revision</b>						
Details of a single revision of a particular document						
revisionID	System ID of the revision, must be unique amongst all revisions	PK xs:string	M	1	1	
revisionRef		xs:string	O	0	1	
revisionNotes		xs:string	O	0	1	
authorID or authorName	Reference to the author who created the revision. Select one of these elements depending if the author is a known user within the system. 'authorID' is a link to a known user specified within the XML export document, while 'authorName' specifies a person outside of the system.	FK xs:string or xs:string	O	0	1	
dateAuthored	The date which the revision was created, could be different from the date the revision was added to the system.	xs:date	O	0	1	
revisionStatus	The status for which the revision has been issued. The data included within this element must be one of the revisionStatuses which has been defined in the XML export document.	xs:string	O	0	1	
status	Indicates the current status of the revision either 'active' or 'deleted'	ref to status	M	1	1	
isLatestRevision	A flag that indicates that this revision is the current revision of the document.	xs:boolean	O	0	1	
createdByID	Link to user who created the revision on the system	FK xs:string	M	1	1	
dateCreated	The date and time that the revision was created on the system	xs:dateTime	M	1	1	
lastModifiedByID	Link to the last user who modified the revision on the system	FK xs:string	O	0	1	
dateLastModified	The date and time that the revision was last modified on the system	xs:dateTime	O	0	1	
##Other	Allows additional revision metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*	
acl	Access Control List for the revision.		O	0	1	
files	Container that holds all the files associated with this revision.		M	1	1	
recipients	A container holding all of the recipients of a revision.		O	0	1	
revisionLinks	A container that holds all the revisionLink elements.		O	0	1	
auditTrail	Container that holds all actions that have been performed of this document.		O	0	1	

<b>files</b>						
A container that holds all the files associated with a particular revision. (one to many files)						
file	One file that is part of a revision.		M	1	*	

<b>file</b>						
Details about a file that is part of a revision, does not include the file data, only information about the file and its location on an external file store.						
fileID	System ID of the file, must be unique amongst all files	PK xs:string	M	1	1	
fileName	Original filename.	xs:string	M	1	1	
MIMEType	MIME Type of the file.	xs:string	O	0	1	
fileSize	The size of the file in bytes.	xs:integer	O	0	1	
status	Indicates the current status of the file either 'active' or 'deleted'	ref to status	M	1	1	
pathToFile	A relative file name and path to the file on disk. Assumes that a local root folder is set up during Import and Export. File name conventions left to exporter.	xs:string	M	1	1	
auditTrail	Container that holds all actions that have been performed of this document.		O	0	1	
##Other	Allows additional file metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*	

<b>recipients</b>						
A container that holds all of the recipients of a particular revision.						
recipient	Organisation, group or user that has been sent this revision.		M	1	*	

## Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

<b>recipient</b>						
User, group or organisation on the distribution list for this particular revision.						
recipientID		PK xs:string	M	1	1	
userID, groupID or organisationID	System ID of the recipient, must match one of the defined ID specified in the XML document.	FK xs:string	M	1	1	
issuedBy	Link to the user who issued the revision to this user, group or organisation.	FK xs:string	M	1	1	
dateIssued	The date and time that the revision was distributed to the user, group or organisation.	xs:dateTime	O	0	1	
dateAcknowledged	The date and time that the user, group or organisation acknowledged the receipt of the revision	xs:dateTime	O	0	1	
status	Indicates the current status of the recipient either 'active' or 'deleted'	ref to status	M	1	1	
##Other	Allows additional recipient metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*	

<b>revisionLinks</b>						
A container that holds all the revisionLink elements.						
revisionLink	Enables Association/Attachments/XREFs between revisions.		M	1	*	

<b>revisionLink</b>						
Enables Association/Attachments/XREFs between revisions.						
revisionLinkID	System ID of the revisionLink, must be unique amongst all revisionLinks.	PK xs:string	M	1	1	
childRevisionID	ID that identifies the revision that is linked to this revision	FK xs:string	M	1	1	
status	Indicates the current status of the link either 'active' or 'deleted'	ref to status	M	1	1	

<b>alias</b>						
Link to document that is contained within another folder, allows the same document to be in more than one folder.						
aliasID	System ID of the alias.	PK xs:string	M	1	1	
name	Display name for this shortcut	xs:string	M	1	1	
documentID	Link to document that the alias represents, must be defined in the export document.	FK xs:string	M	1	1	
##Other	Allows additional recipient metadata to be included in any project export. Additional elements must be defined in additional schemas that are referenced in any XML export document.		O	0	*	

**APPENDIX F BULK EXPORT AND IMPORT  
CHECKLISTS**

**NCCTP STANDARD**  
**INTERNAL TESTING – EXPORT**

**V1.5**

**31ST MARCH 2005**

**CREATED BY: SCOTT MOSES & TIM COLE**

**LAST REVISED BY: SCOTT MOSES**

This set of tests is to be performed on a representative export file generated by the application and is intended to check that each element defined by the NCCTP Standard holds the correct project data.

Any of the tests that apply to elements that are not supported by a vendors export application should be marked in the results table as "Not Supported".

**Definitions:**

<b>ACL</b>	Application Control List
<b>XML Export Data</b>	The data exported according to the NCCTP Standard



Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

VERSION	AUTHOR	ORGANISATION	DATE	NOTES	DISTRIBUTION LIST
1.4	Scott Moses & Tim Cole	Causeway	17 <sup>th</sup> January 2005	First Version of the internal testing documentation to be completed by the exporters of an NCCTP compliant project.	Distributed via the NCCTP web group
1.5	Scott Moses	Causeway	31 <sup>st</sup> March 2005	Changes to testing document requested at the NCCTP technical meeting (13 March 2005). To be circulated to the group for comment.	Distributed via the NCCTP web group.

## Test Elements

### 1. Project ACL

Check that the project ACL is an accurate representation of the permissions of users at the project level.

**TEST:**

*Select at least 2 users and confirm that the project ACL permissions created in the XML Export document for those users are an accurate representation of those that existed in the exporting system.*

Note: these permissions will be cascaded down through the folder structure until new ACL's are defined.

### 2. Folder ACL

For a number of XML folder elements with ACL's defined, check that the individual user/group/organisation permissions are an accurate representation of the systems.

**TEST:**

*Select at least 5 folders with ACL's defined and confirm all individual user/group/organisation permissions in the XML Output Data are accurate representation of the applications ACL for that object.*

Note: these permissions will be cascaded down through child objects (Folders/Documents) until a new ACL is defined.

### 3. Document ACL

For a number of XML Document elements with ACL's defined, check that the individual user/group/organisation permissions are an accurate representation of the systems.

**TEST:**

*Select at least 5 document elements with ACL's defined and confirm all permissions in the XML Output Data are an accurate representation of those that existed in the application.*

Note: these permissions will be copied down to revisions unless new ACL's are defined on the revision.

#### **4. Revision ACL**

For a number of XML revision elements which have ACL's defined, check that the individual user/group/organisation permissions are an accurate representation of the systems.

**TEST:**

*Select at least 5 revision elements that have an ACL defined and confirm that the permissions in the XML Output Data are an accurate representation of those that existed in the exporting application.*

#### **5. Document Audit History**

For a number of documents check that the exported audit history is the same as the document audit history on the system.

**TEST:**

*Select at least 5 documents from the initial application data, locate these in the exported file and confirm that the audit history has been accurately represented in XML Output Data.*

#### **6. Revision Audit History**

For a number of revisions check that the exported audit history is the same as the revision audit history on the system.

**TEST:**

*Select at least 5 documents from the initial application data (can be the same as used in the previous test), locate these in the exported file and confirm that the revision audit history in the XML Output Data is an accurate representation of that which existed in the application.*

#### **7. File Audit History**

For a number of files check that the exported audit history is the same as the file audit history on the system.

**TEST:**

*Select at least 5 files from the initial application data, locate these in the exported file and confirm that the audit history created in the XML Output Data is an accurate representation of that which existed in the application.*

#### **8. Folders**

Check that the total number of folders in the XML Export Data are the same as the number as in the collaboration system.

**TEST:**

*Check the number of folders in the original application and confirm that the same number exists in the XML Output Data as existed in the application.*

## **9. Documents**

Check that the total number of documents in the XML Export Data is the same as the number as in the collaboration system.

### **TEST:**

*Check the number of documents in the original application and confirm that the same number exists in the XML Output Data as existed in the application.*

## **10. Revisions**

Check that the correct number of revisions exist for documents in the XML Export Data(s).

### **TEST:**

*Select at least 5 documents from the initial application data (can be the same as used in previous tests), locate these in the exported file and confirm that the correct number of revisions exist in the XML Output Data as existed in the application.*

## **11. Files**

Check that the correct number of files exist for revisions in the XML Export Data(s).

### **TEST:**

*Select at least 5 document revisions from the initial application data (can be the same as used in previous tests), locate these in the exported file and confirm that the correct number of files exist in the XML Output Data as existed in the application.*

## **12. Users**

Check that the correct number of users is defined within the XML Export Data.

### **TEST:**

*Check the number of users, associated with the project in the original application and confirm that the same number exists in the XML Output Data as existed in the application.*

**Note:** This is of particular importance to vendors that may hold multiple projects each with their own set of distinct users.

### **13. Groups**

Check that the total number of groups in the XML Export Data is the same as the number as in the collaboration system.

**TEST:**

*Check the number of groups, associated with the project in the original application and confirm that the same number exists in the XML Export Data as existed in the application.*

### **14. Group Members**

Check that groups contain the same members in the XML Export Data as they did in the collaborative system.

**TEST:**

*Select at least 3 group elements and confirm that the members are the same in the XML Export Data as existed in the application.*

### **15. Organisations**

Check that the number of organisations that are exported matches the organisations that are held on the system.

**TEST:**

*Check the number of organisations, associated with the project in the original application and confirm that the same number exists in the XML Output Data as existed in the application.*

### **16. Organisation Users**

Check that the list of users for organisations is correct, for the project that is being exported.

**TEST:**

*Select at least 2 organisations, associated with the project in the original application and confirm that the users of each organisation in the XML Output Data are correct. Only those users from the organisation that is associated with the exported project should be included in the export document.*

**Note:** This is a key item for vendors that hold multiple projects, with the same organisations working on multiple projects.

### **17. Correct Current Revision**

Check that the current revision indicated in the XML Export Data match those that are held on the system.

**TEST:**

*Select at least 5 document elements that hold multiple revisions and confirm that the current revision indicated in the application matches that in the XML Data export.*

### **18. Alias Links**

For a number of Alias' check that the documents referenced in the XML Export Data matches those which exist within the system.

**TEST:**

*Select at least 3 alias' from the initial application data, locate these in the exported file and confirm that the document referenced is the same in the XML Output Data as in the application.*

### **19. Revision Links**

Check that the documents are correctly linked to their revisions.

**TEST:**

*Select at least 3 revision links from the originating application, locate these in the exported file and confirm that the document referenced is the same in the XML Output Data as existed in the application.*

### **20. Revision Recipients**

Check that the Revision Recipients associated with any revision in the Exported XML Data accurately match those that exist in the application.

**TEST:**

*Select at least 3 revisions, which have revision recipients from the initial application data, locate these in the exported file and confirm that the document referenced is the same in the XML Output Data as existed in the application.*

### **21. Organisations Offices**

Check that the XML Export Data contains an accurate representation of the office data that existed in the application.

**TEST:**

*Select at least 3 organisations that have at least one office defined from the initial application, locate these in the exported file and confirm that the data in the XML Output Document is an accurate representation of that which existed in the application.*

**22. Correct Head Office**

Check that organisations head offices indicated in the XML Export Data match those that are held on the system.

*TEST:*

*Select at least 2 organisation elements that hold multiple offices and confirm that the head office indicated matches in both the exported XML Data and application.*

**23. Project Address**

Check that the data contained within the project address in the Export Document matches the project address data held on the system.

*TEST:*

*Check the address data, associated with the project in the original application and confirm that the data has been reproduced with no data loss in the XML Output Document.*

**24. User Address**

Check that User Address details have been correctly reproduced in the XML Export Data.

*TEST:*

*Select at least 3 user elements that hold addresses and confirm that the equivalent data exists in the XML Output Data as existed in the application.*

## NCCTP Standard - Export Testing Checklist

Product:

Date of Testing:

No.	Test	Pass/Fail	Comments
1	Check Project ACL		
2	Check Folder ACL		
3	Check Document ACL		
4	Check Revision ACL		
5	Check Document Audit History		
6	Check Revision Audit History		
7	Check File Audit History		
8	Correct Number of Folders		
9	Correct Number of Documents		
10	Documents Hold Correct Number of Revisions		
11	Revisions Hold Correct Number of Files		
12	Check Users		
13	Check Groups		
14	Groups hold the Correct Members		
15	Check Organisations		
16	Check Organisations Users		
17	Check that Current Versions are indicated		
18	Check Alias' refer to the Correct Document		
19	Check Revision's, Revision Links		
20	Check Revision's, Revision Recipients		
21	Check Organisations Offices		
22	Check that Head Offices are Indicated		
23	Project Address		
24	User Address		

**NCCTP STANDARD**  
**INTERNAL TESTING - IMPORT**  
**V1.5**  
**31ST MARCH 2005**

**CREATED BY: SCOTT MOSES & TIM COLE**

**LAST REVISED BY: SCOTT MOSES**

This set of tests is to be performed on a project that has been presented as a compliant NCCTP XML Data File. The checks are intended to verify that the data has been imported correctly to the new Collaborative System ("the application").

The file used to undertake Import testing must be one that has been through Export testing in accordance with the "Internal Testing – Export" document. This will identify any issues relating to the creation of the file that may impact on the Import testing (e.g. elements not present or unsupported in the exporting application). Any tests that refer to data that has not been provided by the exporter should be marked in the results table as "Unavailable for Import".

Any of the tests that apply to elements that are not supported by a vendors import application should be marked in the results table as "Not Supported".

**Definitions:**

<b>ACL</b>	Application Control List
<b>XML Import Data</b>	The data exported according to the NCCTP Standard used here as the Import Data into a collaboration system

VERSION	AUTHOR	ORGANISATION	DATE	NOTES	DISTRIBUTION LIST
1.4	Scott Moses & Tim Cole	Causeway	17 <sup>th</sup> January 2005	First Version of the internal testing documentation to be completed by the importer of a NCCTP compliant project.	Distributed via the NCCTP web group
1.5	Scott Moses	Causeway	31 <sup>st</sup> March 2005	Changes to testing document requested at the NCCTP technical meeting (13 March 2005). To be circulated to the group for comment.	Distributed via the NCCTP web group.

## Test Elements

### 1. Project ACL

Check that the project ACL imported into the application accurately represents the permissions of users at the project level in XML Import Data.

**TEST:**

*Select at least 2 users and confirm all ACL permissions, at the project level in the newly created project are an accurate representation of those that are contained in the XML Import Data.*

Note: these permissions should be cascaded down through the folder structure until new ACL's are defined.

### 2. Folder ACL

Check that the individual user/group/organisation permissions are an accurate representation of the data in the XML Import Data.

**TEST:**

*Select at least 5 folders with ACL's defined and confirm all individual user/group/organisation permissions in the newly created project, are an accurate representation of those contained in the XML Import Data.*

Note: these permissions should be cascaded down through child objects (Folders/Documents) until a new ACL is defined.

### 3. Document ACL

Check that the individual user/group/organisation permissions have been accurately recreated in the new system.

**TEST:**

*Select at least 5 document elements with ACL's defined and confirm all permissions created on the documents in the new system are an accurate representation of those in the XML Import Data.*

Note: these permissions should be copied down to revisions unless new ACL's are defined on the revision.

### 4. Revision ACL

Check that the individual user/group/organisation permissions have been accurately recreated in the new system.

**TEST:**

*Select at least 5 revision elements that have their own ACL defined and confirm that those ACL's have been created in the new project with an accurate representation of the data that is held within the XML Import Data*

## **5. Document Audit History**

For a number of documents check that the imported audit history is the same as the document audit history in the XML Import Data.

### **TEST:**

*Select at least 5 documents from the newly imported project, locate these in the XML Import Data and confirm that the audit history has been accurately reproduced, with no data loss in the new project.*

## **6. Revision Audit History**

Check that the imported audit history is the same as the revision audit history in the provided XML Import Data.

### **TEST:**

*Select at least 5 document revisions from the newly imported project (can be the same documents as used in the previous test), locate these in the XML Import Data and confirm that the revision audit history has been accurately reproduced with no data loss.*

## **7. File Audit History**

For a number of files check that the imported audit history is the same as the file's XML Import Data audit history.

### **TEST:**

*Select at least 5 files from the newly imported project, locate these in the provided XML file and confirm that the audit history has been reproduced with no data loss.*

## **8. Folders**

Check that the total number of folders in the XML Import Data is the same as the number now in the Collaboration System.

### **TEST:**

*Check the number of folders in the new application and confirm that the same number existed in the XML Import Data as now exists in the new application.*

## **9. Documents**

Check that the total number of documents in the XML Import Data is the same as the number as are now in the collaboration system.

### **TEST:**

*Check the number of documents in the new project and confirm that the same number existed in the XML Import Data as now exist in the application.*

### **10. Revisions**

Check that the correct number of revisions exist for documents in the application.

**TEST:**

*Select at least 5 documents from the application data (can be the same as used in previous tests), locate these in the XML Import Data and confirm that the correct number of revisions exist in both.*

### **11. Files**

Check that the correct number of files exist for revisions in the application.

**TEST:**

*Select at least 5 document revisions from the application data (can be the same as used in previous tests), locate these in the XML Import Data and confirm that the correct number of files exist in both.*

### **12. Users**

Check that the correct number of users is defined within the new project as existed in the XML Import Data.

**TEST:**

*Check the number of users, associated with the project in the new application and confirm that the same number existed in the XML Import Data.*

Note: This is of particular importance to vendors that may hold multiple projects each with their own set of distinct users.

### **13. Groups**

Check that the total number of groups in the XML Import Data is the same as the number that are now in the collaboration system.

**TEST:**

*Check the number of groups, associated with the project in the new application and confirm that the same number exists in the XML Import Data.*

### **14. Group Members**

Check that groups contain the same members in the newly imported project as they did in the XML Import Data.

**TEST:**

*Select at least 3 group elements and confirm that the members are the same in the output XML Import Data as now exist in the application.*

### **15. Organisations**

Check that the number of organisations in the XML Import Data matches the organisations that are held on the new system.

**TEST:**

*Check the number of organisations, associated with the new project in the application and confirm that the same number existed in the XML Import Data.*

### **16. Organisation Users**

Check that the list of users for organisations is correct.

**TEST:**

*Select at least 2 organisations from within the imported project and confirm that all their users have been accurately recreated, from the data contained within the XML Import Document.*

Note: This is a key item for vendors that hold multiple projects, with the same organisations potentially working on multiple projects.

### **17. Correct Current Revision**

Check that current revision indicated in the XML Import Data match those that are now held for documents in the project.

**TEST:**

*Select at least 5 documents that hold multiple revisions from within the application and confirm that the current revision indicated matches that in the XML Import Data.*

### **18. Alias Links**

Check that the document referenced by an alias in the XML Import Data matches that which now exists within the system.

**TEST:**

*Select at least 3 alias' from the newly imported project, locate these in the XML Import Data and confirm that the document referenced is the same as now exists in the application.*

### **19. Revision Links**

Check for document revisions that have revision links that these links refer to the correct documents.

**TEST:**

*Select at least 3 revisions, which have revision links in the newly imported project, locate these in the XML Import Data and confirm that all the documents referenced are the same as now exist in the application.*

## **20. Revision Recipients**

Check that the Revision Recipients associated with any revision in the XML Import Data accurately match those that now exist in the application.

### **TEST:**

*Select at least 3 revisions, which have revision recipients from the newly imported project, locate these in the XML Import Data and confirm that all the users referenced are the same as now exist in the application.*

## **21. Organisations Offices**

Check that the newly imported project data contains an accurate representation of the office data that exists in the XML Import Data.

### **TEST:**

*Select at least 3 organisations that have at least one office defined in the XML Import Data, locate these organisations offices in the new system and confirm that the data is an accurate representation of that in the XML Import Document.*

## **22. Correct Head Office**

Check that organisations head offices indicated in the XML Import Data match those that now exist in the application.

### **TEST:**

*Select at least 3 organisation elements that hold multiple offices and confirm that the head office indicated matches in both the XML Import Data and application.*

## **23. Project Address**

Check that the data contained within the project address in the provided XML Import Data matches the project address data that now exists in the application.

### **TEST:**

*Check the address data, associated with the project in the new application and confirm that this information has been reproduced with no data loss from that contained within the XML Import Document.*

## **24. User Address**

Check that the user address information has been correctly reproduced in the new application.

### **TEST:**

*Select at least 3 user elements that hold address information and confirm that this address data in the new system is an accurate representation of that which existed in the XML Import Document Data.*

#### **24. Truncated Fields**

If size limits are imposed by the importing application on any data that can be populated via an NCCTP import process, then the importing user should be made aware that the data will be truncated.

**TEST:**

*Select at least 3 elements for which size restrictions apply in the importing application, and that the elements in the XML Import Document Data exceed those limits. Ensure that the importing utility makes the user aware that the data will/has been truncated by the application.*

**Note:** The test is passed if the receiving application either

- Clearly warns the user of the specific instances where truncation will take place and prevents duplicate string elements resulting; and/or,
- Provides the user with clear visibility of the corrective action that is to be taken and the means to avoid duplicate strings.

**NCCTP Standard - Import Testing Checklist**

Product:

Date of Testing:

No.	Test	Pass/Fail	Comments
1	Check Project ACL		
2	Check Folder ACL		
3	Check Document ACL		
4	Check Revision ACL		
5	Check Document Audit History		
6	Check Revision Audit History		
7	Check File Audit History		
8	Correct Number of Folders		
9	Correct Number of Documents		
10	Documents Hold Correct Number of Revisions		
11	Revisions Hold Correct Number of Files		
12	Check Users		
13	Check Groups		
14	Groups hold the Correct Members		
15	Check Organisations		
16	Check Organisations Users		
17	Check that Current Versions are indicated		
18	Check Alias' refer to the Correct Document		
19	Check Revision's, Revision Links		
20	Check Revision's, Revision Recipients		
21	Check Organisations Offices		
22	Check that Head Offices are Indicated		
23	Project Address		
24	User Address		
25	Truncated Fields		



## **APPENDIX G INDEPENDENT TESTING OF NCCTP IMPLEMENTATIONS**



**NCCTP STANDARD**

**INDEPENDENT TESTING PROPOSAL DOCUMENT**

**V1.2**

**4TH APRIL 2006**

**CREATED BY: SCOTT MOSES**

This document outlines the procedure that should be followed by the vendor and tester when an organisations NCCTP implementation is independently tested.



Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

VERSION	AUTHOR	ORGANISATION	DATE	NOTES	DISTRIBUTION LIST
1.0	Scott Moses	Causeway	15 <sup>th</sup> September 2005	Draft Version of the proposal for independent testing of each organisation implementation of the NCCTP data exchange standard	Distributed via the NCCTP web group
1.1	Scott Moses	Causeway	15 <sup>th</sup> March 2005	Additional of Draft forms to complete during the testing process.	Distributed via the NCCTP web group
1.2	Scott Moses	Causeway	4 <sup>th</sup> April 2005	Updated for 29/03/2006 meeting. New Import test included.	Distributed via the NCCTP web group

## **Independent Testing of NCCTP Implementations**

### **Activities to be completed prior to the commencement of Verification**

The prospective organisation must have submitted a sample project that complies with the current supported version of the NCCTP Data Exchange Standard, along with supporting material<sup>iv</sup> that helps in the interpretation of the project exported. The submission must also include completed documentation that states the export and subsequent import completed successfully; and the areas of the Data Exchange Standard supported during each of these activities.

Activity	Completed
Successfully Exported Project	
Exported Project Supporting Material	
Completed Export Testing Documentation	
Successfully Import Project	
Completed Import Testing Documentation	
Documentation of Procedures <sup>v</sup>	

### **Verifying the Data Exported to the NCCTP Standard**

#### *Minimum Requirements of the Exported Sample Project*

- The project must contain at least one user who has participated but has now had their access removed.
- The project must contain one and only one project object, no sub-projects are allowed in the export file.
- Exported projects must contain a minimum of 5 users who exist within at least 3 different organisations.
- Access Control Lists used by the objects within the exported project should utilise a least one of the three levels of permissions allowed Document, Revision or File.
- The selected project should contain the maximum amount of content that is supported by the NCCTP Standard. Any of the elements that are not included in the export should be considered not supported by the source application. To help evaluate the elements contained in the export the tester should make reference to the completed internal testing documentation provided by each company.

---

<sup>iv</sup> Supporting Material may consist of project metrics, screenshots of each of the critical sections for testing, and or a viewable archive of the project that includes access to all data exported.

<sup>v</sup> Each Organisation should supply details of how they resolve anomalies between the data that they importing and their own system.

### *Scalability Testing of the Export Utility*

The organisation must prove that their extraction application is capable of exporting data from large projects that better replicate those that actually exist on the live systems deployed by the vendor. The organisation is not required to export a client's project from their live system if they do not wish to for any particular reason, and can provide their own large test project for this check. However the project must be installed on one of the live system operated by the provider. The scalability test is not designed to check the NCCTP functionality supported by the vendor's application just that the application is capable of extracting real project data. Therefore the minimum requirements of the project used for scalability testing are:

- The project must contain a minimum of 10,000 documents
- The exported data must validate against the NCCTP Data Exchange Standard

### *Testing Procedure for Verification of the Export Utility*

1. **Vendor** supplies access to the project data contained within the system through their standard interface.
2. **Tester** verifies that the project selected for export contains all of the different elements that the organisation claims to support, by examining the export testing documentation supplied.
3. The **vendor** conducts the export of the sample project.
4. The **tester** should then validate the exported project against the current version of the NCCTP Data Exchange Standard<sup>vi</sup>. If the project was exported using the xml file splitting supported by the standard then a selection of the separate document xml files should also be validated against the standard.
5. The **tester** should then complete a new instance of NCCTP Export Test Documentation for the project that has been exported. A successful test should return no fails, in any of the areas supported by vendors system.
6. Either prior, during or after the sample project has been exported and tested; the **vendor** should allow the **tester** access to large project that is to be used for scalability testing on the live system.
7. The **vendor** then needs to start the export of the project used for scalability testing.
8. Once export is completed the project should be validated by the **tester** against the current version of the NCCTP Data Exchange Standard using the tool specified by the NCCTP Technical Group.
9. The **tester** should then conduct some basic tests on the xml documents produced to satisfy themselves that the exported data is OK.
10. The **tester** completes a report on how the testing has gone stating if the vendor organisation has successfully completed this stage of the

---

<sup>vi</sup> The NCCTP Technical Group needs to specify a tool to be used by the person conducting the tests to validate the exported project data.

test. If the export test has not been successfully completed then the **tester** should indicate the areas where the vendor has failed to meet the stated requirements.

11. If the test has been failed then arrangements need to be made between the vendor and tester to re run the tests at a later date.

### *Proposed Export Testing Plan (Day 1)*

Activity	09:00	09:15	09:30	09:45	10:00	10:15	10:30	10:45	11:00	11:15	11:30	11:45	12:00	Lunch
1 - Tester Examines Sample Project														
2 - Tester Verifies that Sample Project meets Requirements														
3 - Sample Project Data is Exported														
4 - Exported Data Validated Against Schema														
5 - NCCTP Export Testing Documentation Completed														
6 - Tester Examines the Project to be used for Scalability Testing														
7 - Scalability Testing project is exported														
8 - Scalability Testing Project Export Data Validated Against Schema														
9 - Selected Tests Against Project														
10 - Tester writes a export summary report														

### *NCCTP Supported Export Extensions Testing*

The NCCTP Standard has been designed with extensibility in mind and member organisations are encouraged to enhance their exported data with other data that they hold that is not directly supported by the standard. Although for an organisation to claim compliance with the standard these extensions do not need to be included or tested, the person conducting the test should at a minimum confirm that the extended export still validates against the current version of the standard using the validation tool specified. If an organisation provides extensions to the schema that do validate against the standard then these can be recorded as supporting an extended version of the NCCTP standard.

## **Verifying the Data Imported from the NCCTP Standard**

### *Project Data Required for the Import Tests*

The import test should be conducted with the sample project held by the NCCTP that contains all of the elements contained within the NCCTP schema. To aid with the verification of the data imported into the system the tester can utilise the supporting material that is supplied with the sample project.

### *Extracted Project Data*

In addition to the import test conducted with the example project that contains all NCCTP elements, the tester should select at random one of the compliant projects exported by any of the system. The list of available project for selection by the tester will include the latest compliant project export form each NCCTP organisation, which has been verified as compliant by other NCCTP members.

### *Scalability Testing of the Import Utility*

For a vendor to claim that they are fully NCCTP compliant they must demonstrate that their import application is fully scalable, and capable of importing projects that are more realistic representations of the projects that could potentially be moved. The importing vendor is not expected to import a project that has been exported from one of the other compliant systems, but to use the standard example scalability testing project that is held centrally by the NCCTP Technical group. The scalability import test is not designed to check the NCCTP functionality supported by the vendor's application just that the application is capable of importing project data of any magnitude.

Therefore the minimum requirements of the project used for scalability testing are:

- The project to be imported must contain a minimum of 10,000 documents
- The project data must validate against the NCCTP Data Exchange Standard

## Testing Procedure for Verification of the Import Utility

1. **Tester** supplies the **vendor** with the sample project, taken from the NCCTP Technical group's central repository to ensure that no pre-processing and/or data manipulation has occurred.
2. **Vendor** examines the project and conducts any pre-processing required to prepare the data for import.
3. **Vendor** imports the data into one of their systems<sup>vii</sup>.
4. The **tester** should complete a new instance of NCCTP Import Test Documentation for the project that has been imported. A successful test should return no fails, in any of the areas supported by **vendor's** system<sup>viii</sup>.
5. **Tester** supplies the vendor with a sample project that is suitable for conducting scalability testing of the project import utility.
6. **Vendor** examines the project and conducts any pre-processing required to prepare the data for import.
7. **Vendor** then imports the large project onto one of their systems.
8. The **tester** then verifies that the project has been imported correctly by conducting a series of tests<sup>ix</sup>.
9. The **tester** completes a report on how the testing has gone stating if the vendor organisation has successfully completed this stage of the assessment. If the import test has not been successfully completed then the **tester** should indicate the areas where the **vendor** has failed to meet the requirements.
10. **Tester** selects which existing project exported by another NCCTP Member that the organisation will import.
11. The **Vendor** examines the project and conducts any pre-processing required to prepare the data for import.
12. **Vendor** Imports the project data into one of their systems.
13. The **tester** should complete a new instance of NCCTP Import Test Documentation for the project that has been imported. A successful test should return no fails, in any of the areas supported by **vendor's** system.
14. **Tester** writes the import summary report.
15. **Tester** concludes the Independent testing process and completes any outstanding documentation.
16. If the test has been failed then arrangements need to be made between the **vendor** and **tester** to re run the tests at a later date.

---

<sup>vii</sup> The NCCTP Technical Group will need to specify if the project must be imported into a live system, or if the project can be imported into a test system. The person conducting the test needs to ensure that the import been conducted is creating the project that they are due to carry out there tests on.

<sup>viii</sup> Please note that an understanding of how a system import application deals with a particular situation will be needed. The vendor should document the different ways in which potential problems are overcome. For example how a system based on document level permissions will deal with importing a project at includes permissions stored at the version level.

<sup>ix</sup> The number and type of tests that need to be conducted on the imported project used for scalability testing need to be collectively defined by the NCCTP Technical Group.

# Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

## Proposed Import Testing Plan (Day 1)

Activity	12:00	12:15	12:30	12:45	13:00	13:15	13:30	13:45	14:00	14:15	14:30	14:45	15:00	15:15	15:30	15:45	16:00	16:15	16:30	16:45	
1 - Project Data Supplied to Vendor by Tester																					
2 - Vendor Examines and Manipulates Project Data																					
3 - Vendor Imports the Project into Collaborative System																					
4 - NCCTP Import Testing Documentation Completed																					
5 - Tester supplies the project data for scalability testing																					
6 - Vendor Prepares the project to be used for scalability testing																					
7 - Scalability Testing Project Imported																					
8 - Selected Tests against Project																					
9 - Tester writes a import summary report																					

## Proposed Import Testing Plan (Day 2)

Activity	09:00	09:15	09:30	09:45	10:00	10:15	10:30	10:45	11:00	11:15	11:30	11:45	12:00	12:15	12:30	12:45	13:00	13:15	13:30	13:45	
10 - Tester selects which existing project the organization will import																					
11 - Vendor Examines and Manipulates Project Data																					
12 - Vendor Imports the Project into their Collaborative System																					
13 - NCCTP Import test Documentation Completed																					
14 - Tester writes a import summary report																					
15 - Tester Concludes Independent Testing																					

## NCCTP Supported Export Extensions Testing

The tester will be unable to check any extensions exported from any of the systems and therefore this will not be any part of the test.

## **NCCTP Independent Testing Forms**

**Form Name:** NCCTP Independent Testing Request Form

**Form:** NCCTP-IT#1-REQ

**Completed By:** Vendor

**When:** After fully implementing the standard and completing the associated documentation

**Form Name:** NCCTP Independent Testing Pre-Test Checks Form

**Form:** NCCTP-IT#2-PT

**Completed By:** Tester

**When:** Prior to arranging to visit the Company Site

**Form Name:** NCCTP Independent Testing Export Checks Form

**Form:** NCCTP-IT#3-EXP

**Completed By:** Tester

**When:** While Conducting the Export Checks on Site

**Form Name:** NCCTP Independent Testing Import Checks Form

**Form:** NCCTP-IT#4-IMP

**Completed By:** Tester

**When:** While Conducting the Import Checks on Site

**Form Name:** NCCTP Independent Testing Assessment Report

**Form:** NCCTP-IT#5-REP

**Completed By:** Tester

**When:** After completing the required tests

<b>NCCTP Independent Testing Request Form</b>	
<b>Organisation:</b>	
<b>Organisations Representative:</b>	
<b>Date of Request:</b>	
<b>NCCTP Standard Version:</b>	
<b>1. Our Organisation has implemented the NCCTP Standard and can export project data to it.</b>	<b>Yes / No</b>
<b>1.1 We have posted an example project on the NCCTP Discussion Group.</b>	<b>Yes / No</b>
<b>1.2 We have supplied supporting material with our example exported project.</b>	<b>Yes / No</b>
<b>1.3 We have completed the Internal Testing Documentation for the Export.</b>	<b>Yes / No</b>
<b>2. Our Organisation has implemented the NCCTP Standard and can import project data from it.</b>	<b>Yes / No</b>
<b>2.1 We have completed the Internal Testing Documentation for the Import.</b>	<b>Yes / No</b>
<b>2.2 We have documented our procedures for mapping data that does not conform to our system.</b>	<b>Yes / No</b>

<b>NCCTP Independent Testing Pre-Test Checks Form</b>	
<b>Organisation:</b>	
<b>Organisations Representative:</b>	
<b>Date of Request:</b>	
<b>NCCTP Standard Version:</b>	
<b>1. Our Organisation has implemented the NCCTP Standard and can export project data to it.</b>	<b>Yes / No</b>
<b>1.1 We have posted an example project on the NCCTP Discussion Group.</b>	<b>Yes / No</b>
<b>1.2 We have supplied supporting material with our example exported project.</b>	<b>Yes / No</b>
<b>1.3 We have completed the Internal Testing Documentation for the Export.</b>	<b>Yes / No</b>
<b>2. Our Organisation has implemented the NCCTP Standard and can import project data from it.</b>	<b>Yes / No</b>
<b>2.1 We have completed the Internal Testing Documentation for the import.</b>	<b>Yes / No</b>
<b>2.2 We have documented our procedures for mapping data that does not conform to our system.</b>	<b>Yes / No</b>

## NCCTP Independent Testing Export Checks Form

<b>Organisation:</b>	
<b>Organisations Representative:</b>	
<b>Independent Tester Name(s):</b>	
<b>Date and Time of Test:</b>	
<b>NCCTP Standard Version:</b>	
<b>1. Check Project ACL</b>	<b>Yes / No / Not Supported</b>
<b>2 Check Folder ACL</b>	<b>Yes / No / Not Supported</b>
<b>3 Check Document ACL</b>	<b>Yes / No / Not Supported</b>
<b>4 Check Revision ACL</b>	<b>Yes / No / Not Supported</b>
<b>5. Check Document Audit History</b>	<b>Yes / No / Not Supported</b>
<b>6. Check Revision Audit History</b>	<b>Yes / No / Not Supported</b>
<b>7. Check File Audit History</b>	<b>Yes / No / Not Supported</b>
<b>8. Correct Number of Folders</b>	<b>Yes / No / Not Supported</b>
<b>9. Correct Number of Documents</b>	<b>Yes / No / Not Supported</b>
<b>10. Documents hold the correct number of revisions</b>	<b>Yes / No / Not Supported</b>
<b>11. Revisions hold the correct number of files</b>	<b>Yes / No / Not Supported</b>
<b>12. Check Users</b>	<b>Yes / No / Not Supported</b>
<b>13. Check Groups</b>	<b>Yes / No / Not Supported</b>
<b>14. Groups hold the correct members</b>	<b>Yes / No / Not Supported</b>
<b>15. Check Organisations</b>	<b>Yes / No / Not Supported</b>
<b>16. Check Organisations Users</b>	<b>Yes / No / Not Supported</b>
<b>17. Check current versions are indicated</b>	<b>Yes / No / Not Supported</b>
<b>18. Check Alias' refer to the correct Documents</b>	<b>Yes / No / Not Supported</b>

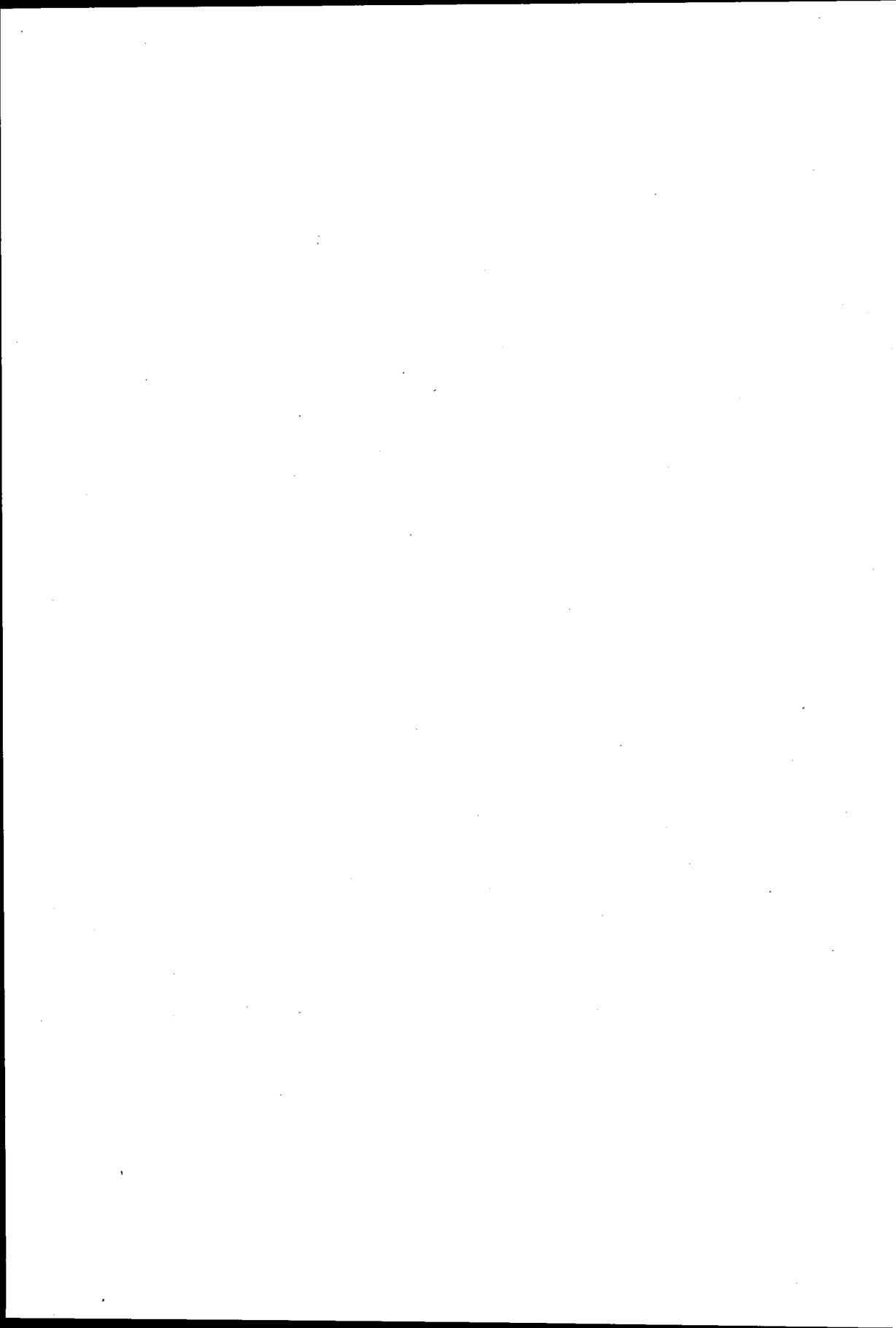
19. Check Revision's, Revision Links	Yes / No / Not Supported
20. Check Revision's, Revision Recipients	Yes / No / Not Supported
21. Check Organisations Offices	Yes / No / Not Supported
22. Check that Head Offices are Indicated	Yes / No / Not Supported
23. Project Address	Yes / No / Not Supported
24. User Address	Yes / No / Not Supported

## NCCTP Independent Testing Import Checks Form

<b>Organisation:</b>	
<b>Organisations Representative:</b>	
<b>Independent Tester Name(s):</b>	
<b>Date and Time of Test:</b>	
<b>NCCTP Standard Version:</b>	
<b>1. Check Project ACL</b>	<b>Yes / No / Not Supported</b>
<b>2. Check Folder ACL</b>	<b>Yes / No / Not Supported</b>
<b>3. Check Document ACL</b>	<b>Yes / No / Not Supported</b>
<b>4. Check Revision ACL</b>	<b>Yes / No / Not Supported</b>
<b>5. Check Document Audit History</b>	<b>Yes / No / Not Supported</b>
<b>6. Check Revision Audit History</b>	<b>Yes / No / Not Supported</b>
<b>7. Check File Audit History</b>	<b>Yes / No / Not Supported</b>
<b>8. Correct Number of Folders</b>	<b>Yes / No / Not Supported</b>
<b>9. Correct Number of Documents</b>	<b>Yes / No / Not Supported</b>
<b>10. Documents hold the correct number of revisions</b>	<b>Yes / No / Not Supported</b>
<b>11. Revisions hold the correct number of files</b>	<b>Yes / No / Not Supported</b>
<b>12. Check Users</b>	<b>Yes / No / Not Supported</b>
<b>13. Check Groups</b>	<b>Yes / No / Not Supported</b>
<b>14. Groups hold the correct members</b>	<b>Yes / No / Not Supported</b>
<b>15. Check Organisations</b>	<b>Yes / No / Not Supported</b>
<b>16. Check Organisations Users</b>	<b>Yes / No / Not Supported</b>
<b>17. Check current versions are indicated</b>	<b>Yes / No / Not Supported</b>
<b>18. Check Alias' refer to the correct Documents</b>	<b>Yes / No / Not Supported</b>

<b>19. Check Revision's, Revision Links</b>	<b>Yes / No / Not Supported</b>
<b>20. Check Revision's, Revision Recipients</b>	<b>Yes / No / Not Supported</b>
<b>21. Check Organisations Offices</b>	<b>Yes / No / Not Supported</b>
<b>22. Check that Head Offices are Indicated</b>	<b>Yes / No / Not Supported</b>
<b>23. Project Address</b>	<b>Yes / No / Not Supported</b>
<b>24. User Address</b>	<b>Yes / No / Not Supported</b>
<b>25. Truncated Fields</b>	<b>Yes / No / Not Supported</b>

**APPENDIX H NCCTP INCREMENTAL DATA  
EXCHANGE PROPOSED STANDARD**

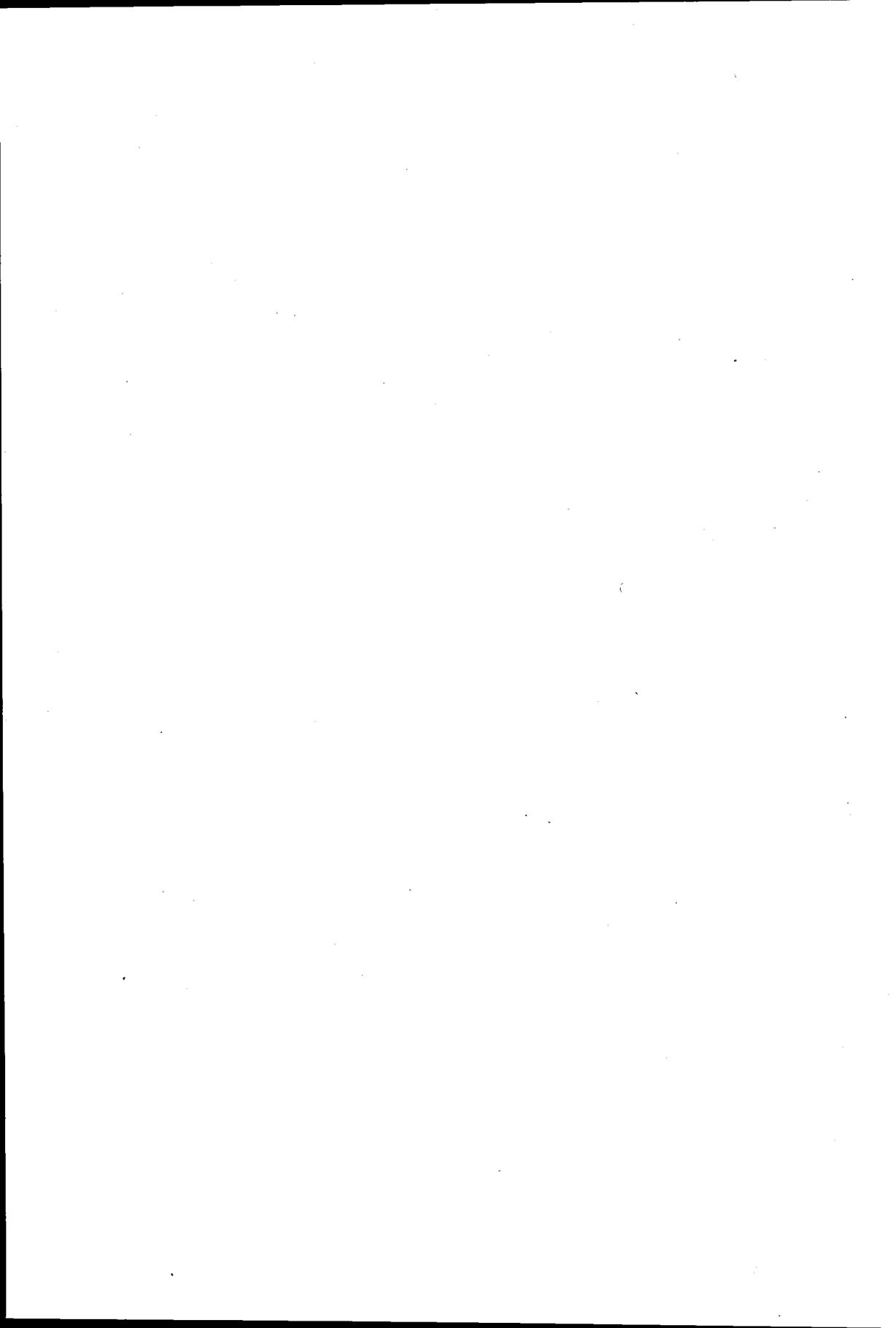


---

NCCTP Incremental  
Project Updating  
Specification [DRAFT]

---

NCCTP Specification Request 2  
Version 1.0  
19 October 2005



## **Acknowledgements**

This specification is the collaborative product of

Scott Moses (specification lead,  
Causeway),

Scott Moses (author, Causeway)

And other members of the NCCTP expert Group, as well as many others who contributed with corrections and suggestions.

**License**

This specification may only currently be used by ncctp member organisations.

---

## 1 Preface

---

This is version 1.0 of the NCCTP Incremental Project Updating Specification.

### **1.1 Documents Included**

---

This specification includes:

- This document in Microsoft Word Format (ncctpIPUv1.102.doc).
- The XML Schema Document that defines the data transferred (IPUv1.0.xsd).

In case of discrepancy between this document and the contents of the IPUv1.0 XML Schema this document should be considered normative.

---

## 2 Introduction

---

### 2.1 Motivation

---

As the number of different extranet providers has increased and more organisations are required to use a multitude of these systems concurrently on numerous different projects, the need incremental data transfer has become apparent. The aim of the ncctp incremental project updating specification is to provide the common framework that would allow project data to be transferred between systems throughout the duration of the project.

Since all systems will export their incremental updates to the same generic format application developers will only need to build a single incremental export and import application for their content repository, allow multiple connection instances.

Clients will be able to bring back information stored in many distributed project extranets to their central enterprise content management system, at key stages during the project. Conversely they will be able to push information from their chosen central repository to any project specific extranet

### 2.2 Goals

---

The guiding principles governing the design of this specification are:

**It should not be tied to any particular underlying architecture, data source or protocol.**

The specification is, of course, built upon the ncctp bulk data exchange standard, which describes the commonalities that exist between different extranet systems used by the construction industry. Hence achieving this goal is not difficult in itself. The main challenge is to ensure that the solution is flexible enough to be used by all different instances of systems in use.

**It should allow for relatively easy implementation on top of as wide a variety of existing systems as possible.**

A concentrated effort was made to ensure that it was relatively easy to implement both the incremental export and import application on top of most major vendor's systems used by the construction industry.

**It should transfer the minimum data between systems, allowing for data consistency to be maintained.**

Incremental project updates could be achieved using the existing ncctp bulk data exchange standard, however this would result in a huge quantity of excess data been extracted and transferred. This specification has reduced this data to the minimum required for successfully maintaining the data consistency between participating systems, while maintaining the project structure defined by the bulk data exchange standard.

---

## 3 Use Cases

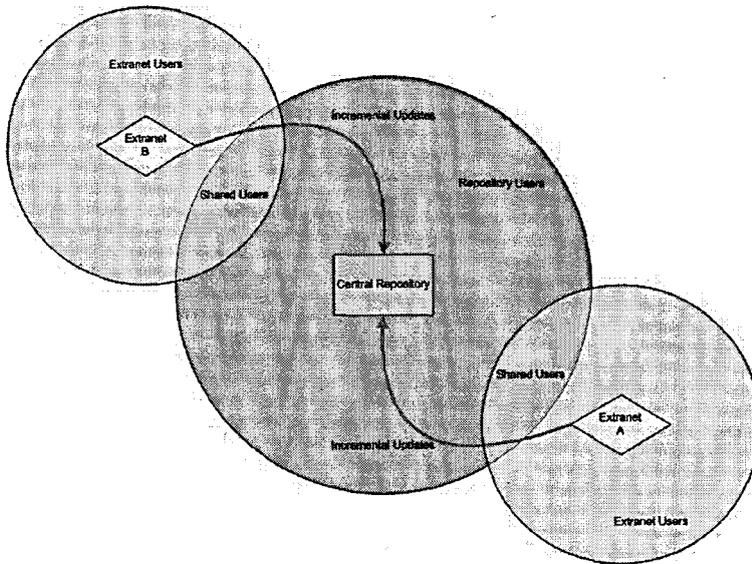
---

### 3.1 Single Central Repository

---

WeBuild Corporation, a large contractor in the construction sector operates an enterprise wide content management system, but also participates on projects that utilise independently host extranet systems. Their Knowledge Management Team would like for all project data to be held within the one system so that all employees can access relevant information.

The team implements the nctp incremental project updating solution and receives regular updates from each project that they are participating on, allowing every member of the organisation access to valuable knowledge without the need to access the extranet system directly.



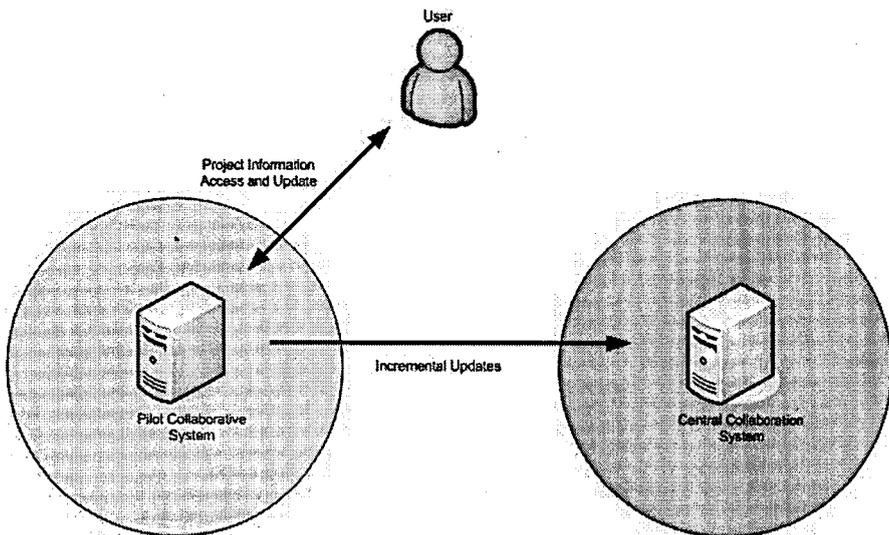
With the incremental updating solution operating successfully, as projects finish there is no need for WeBuild Corporation to loose any of the data as they have been keeping their own system updated throughout. This means that they will not need to ask the client or system provider for relevant project data at completion.

### 3.2 Transfer of Hosting Responsibility

---

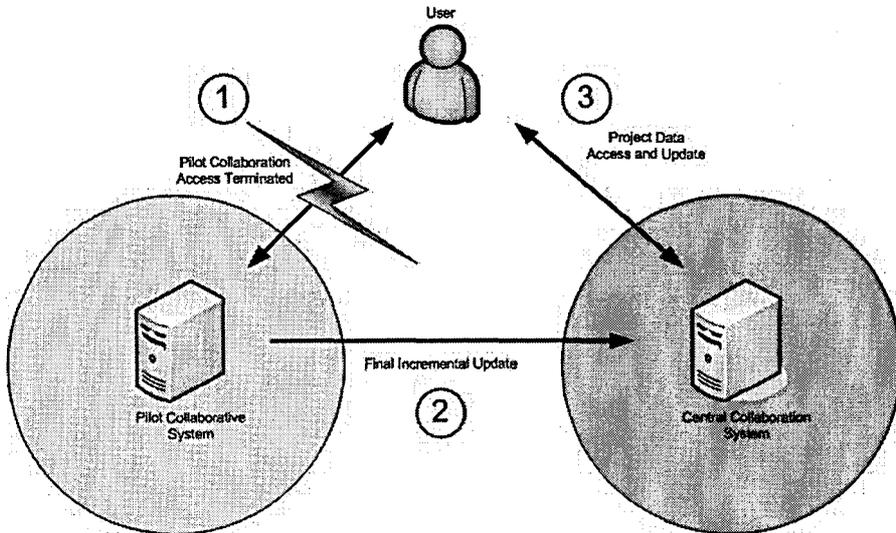
Grand Hotels, a worldwide client who build and operate hotels, have been piloting a number of different extranet systems, to be used throughout the entire enterprise. After a year long assessment one solution has been selected, and the company wish to have all project data stored within the same system.

The projects which are active on the unsuccessful extranet providers are on tight schedules and Grand Hotels do not want to risk any down time or errors that are associated with one time only bulk transfer of information.



They therefore select to use the ncctp incremental project updating solution to start replicating the dispersed project data on to their central collaboration system, while people are still using the extranet system for project collaboration.

Once assured that the data stored on the enterprise system is an accurate representation of that on the extranet system, and all external parties have been given the necessary training for new system, they make the switch between the project been hosted by the extranet provider remotely to the project been hosted internally.



Since incremental project updates can be carried out at short time intervals Grand hotels were able to make the switch one weekend, having full confidence that the data was accurate.

### **3.3 Updating Project Extranet from Central Repository**

Infrastructure Design and Construct are an organisation that specialise in road and rail projects. Although they have a central corporate collaboration system, they also make extensive use of project extranets for the unique advantages a neutral hosted solution brings. Their team would like for data and updates to be automatically pushed from the central to extranets that their employees participate in.

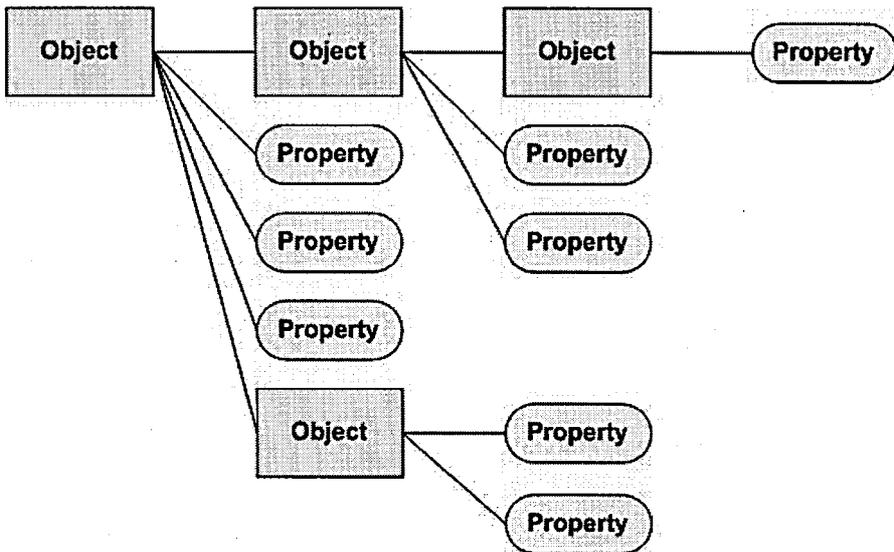
The organisation selects the nctp incremental data exchange methodology as a mechanism for populating and updating the information which is held on external project collaborative platforms. Eliminating the manual need to check and update the information stored.

With the incremental updating solution working correctly 3<sup>rd</sup> party organisations now have access to the latest information without the need to access the central system. While Infrastructure Design and build do not have to check to see which documents need updating on which extranet.

## 4 The Generic Repository Model

---

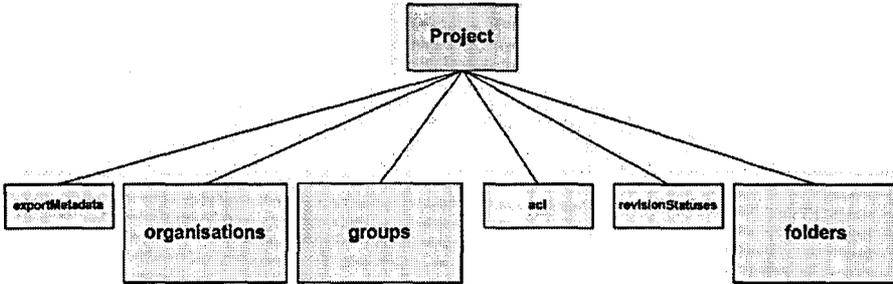
A Content Repository consists one or more projects, each of which contain a series of objects held in a tree structure. Objects can optionally contain other objects but must contain specific properties dependant upon their object type, along with a series of optional properties. Each object and property has only one parent object, and properties cannot contain objects or other properties.



Any object in the hierarchy can be identified from its unique id property and associated object type, note that id's are only guaranteed to be unique within a single object class. For example specifying a Folder object with an id property of 12345 would locate the relevant object in the tree structure.

## 4.1 Major Repository Branches

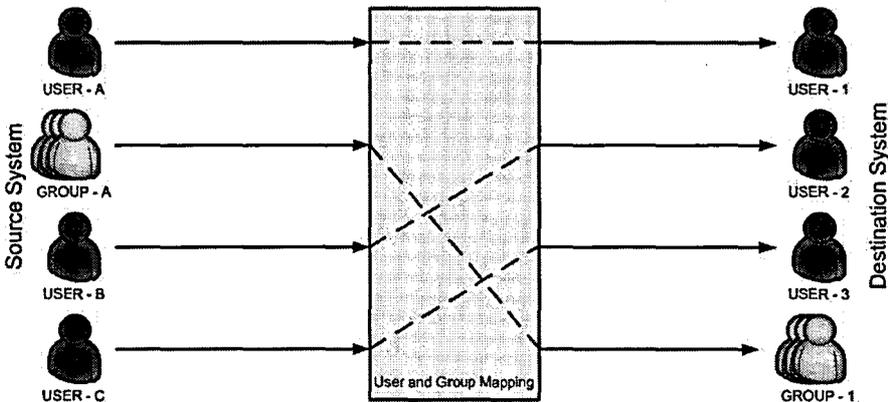
The Content Repository is split into three major branches; the first holds the information relating to project users, the second holds information relating to user collections within the project, while the third stores the project documents.



### 4.1.1 Project Users and Project Groups

For project data to be successfully transferred between systems the same user accounts must exist on all systems. Incremental project updates received from the source system will contain references to user and group objects that exist within the source systems content repository. These references will need to be translated by the destination system to users and groups objects that exist within its content repository.

To allow these translations the destination system is required to keep user and group object mapping data; While the source system is required to maintain that mapping data through the incremental updates which it passes.



#### **4.1.2 Project Documents**

---

Project documents refer to documents and the filing structure in which they are stored in the content repository.

For new object to be created in the correct location of the documents branch of the content repository the same objects must exist on all the participating systems. Incremental project updates received from the source system will contain references to other objects that exist within the source systems content repository. These references will need to be translated by the destination to objects that exist within the destination systems content repository.

To allow for translations to be made the destination system is required to keep object mapping data; while the source system is required to maintain that data through the incremental updates it passes to the destination system.

---

## 5 Incremental Project Updates

---

Incremental project updating is a one way process that allows replication of project data between two different extranet systems. This one way only transfer of data means that only participants on the source system will be able to edit the project objects, and that the data should be read only once in the destination system.

If data within the destination system is modified it may be impossible for the incremental process to find the correct object that it should be working with.

Project updates can take the form of object creation, modification, or deletion only.

However an additional section will cover the phenomena of object translation within the overall ncctp object tree. For Example a user is moved from one organisation to another organisation, or a document is moved from one folder to another folder.

### **5.1 Object Creation**

---

All new objects created in the source system must be included in any incremental project update as created objects.

### **5.1.1 Project Object Creation**

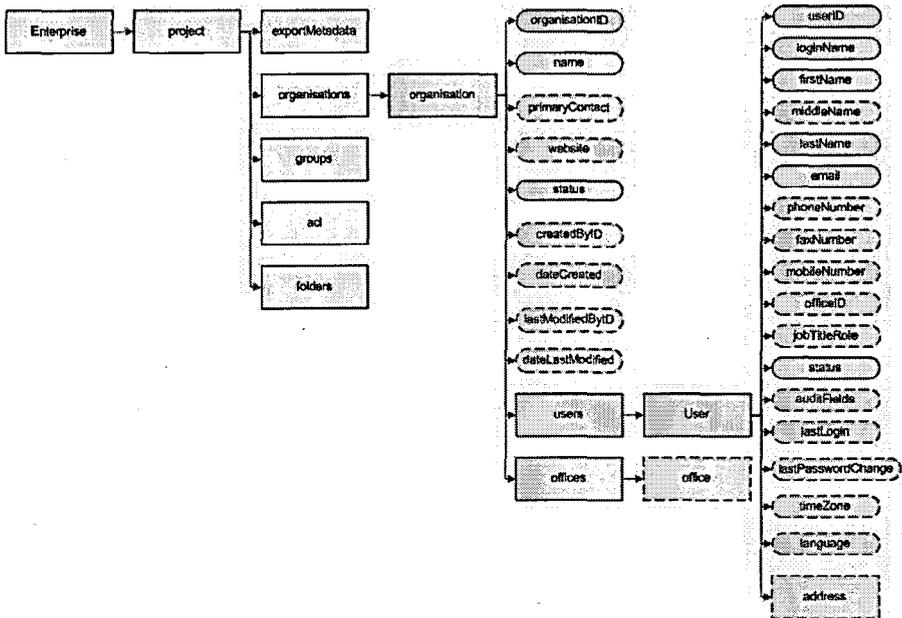
---

Project objects should not be created through the incremental project updating process. The initial transfer of information between the systems should take the form of a single bulk extraction followed by a bulk import of data into the destination system. After the initial bulk export which contained the project object incremental project updates can be created regardless of whether or not the data has been imported into the destination system yet.

All bulk project exports must comply with the latest supported version of the ncctp bulk data exchange schema.

## 5.1.2 Organisation Object Creation

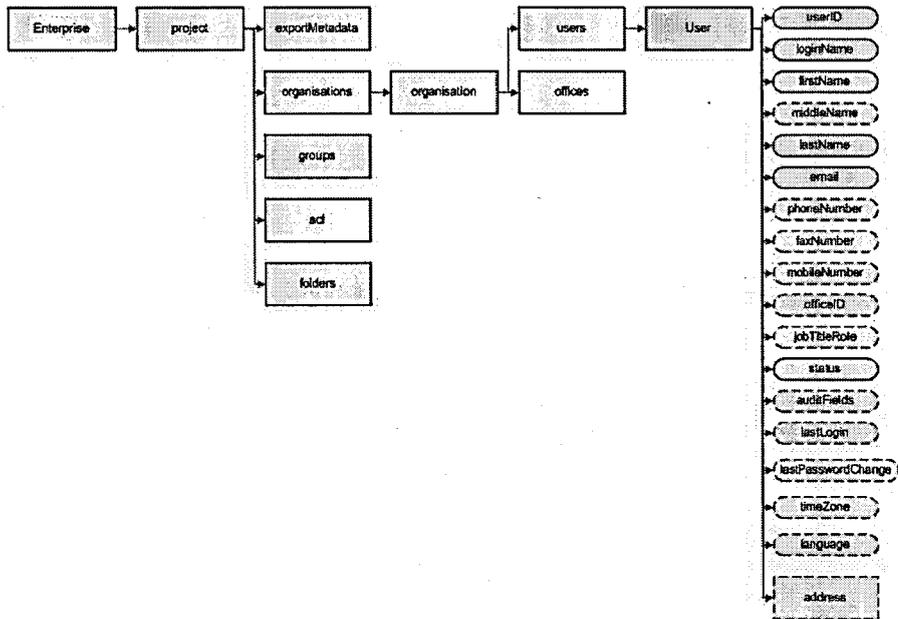
An Organisation is a direct child of the project object, so within the creation definition both the object type project and its child property id must be specified, along with the organisation object definition. The definition of organisation passed within the create update must comply with the latest supported version of the ncctp bulk data exchange standard; allowing both organisation users and offices to be defined in the same incremental update.



The organisationID property of the organisation object must be unique amongst those already synchronised with the destination system; else the incremental creation of the organisation will be considered an update of an existing organisation not the creation of a new organisation.

### 5.1.3 User Object Creation

A user object is a child of an organisation object, so within the creation definition both the organisation type and its associated id property must be specified. The definition of a user passed as part of the create update must comply with the latest supported version of the nccpt bulk data exchange standard.



Only user accounts that previously did not exist as a participant on the project should be passed as part of a create update, those that have changed organisations, or been undeleted should not be created as new objects.

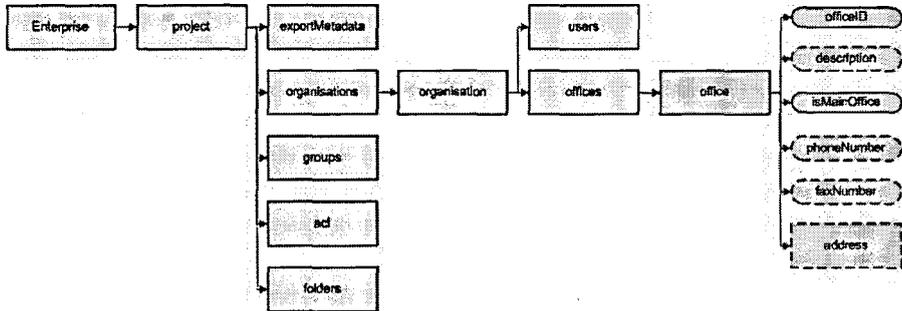
Thus the create update will instruct the destination system to create a new user account on the destination system, that belongs to one of the organisations that are synchronised between the systems.

---

### 5.1.4 Office Object Creation

---

An office object is a child of an organisation object, so the create definition for the office object must include both the organisation type and its associated id property. The definition of an office passed must comply with the latest supported version of the ncctp bulk data exchange standard.



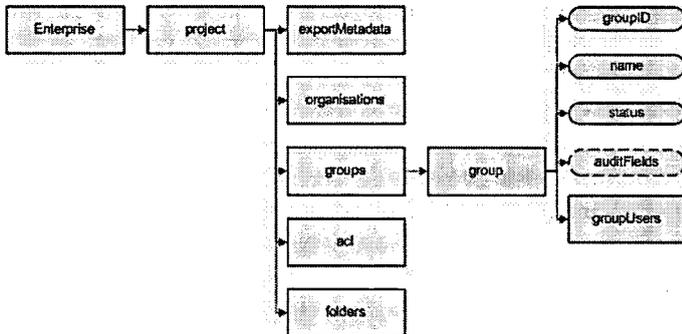
Incremental creation of new office objects will most likely follow the creation of new instances of the organisation object, but can equally be added to organisation objects that have existed since the initial bulk data transfer.

The create update instructs the destination system to make a new instance of the office object as a child of the organisation specified. However this new office must not have property values that conflict with a pre-existing office object. For example the property 'isMainOffice' cannot be set to true for a new office if an office already existing has the property value set to true. Should the new office be required to be the new main office then additional modify updates should be sent to adjust the property values on both office objects.

### 5.1.5 Group Object Creation

---

The group creation incremental update includes all information required to create a group in the destination system, and must comply with the group object definition specified in the nctpt bulk data exchange standard. This group definition must be contained within a single xml document which in its entirety validates against the latest version of the nctpt incremental data exchange standard.



Any group object created must have a unique property value for groupID within the project which is being incrementally replicated. Any group update whose groupID property value matches an existing group will be assumed to be a modification of that existing group.

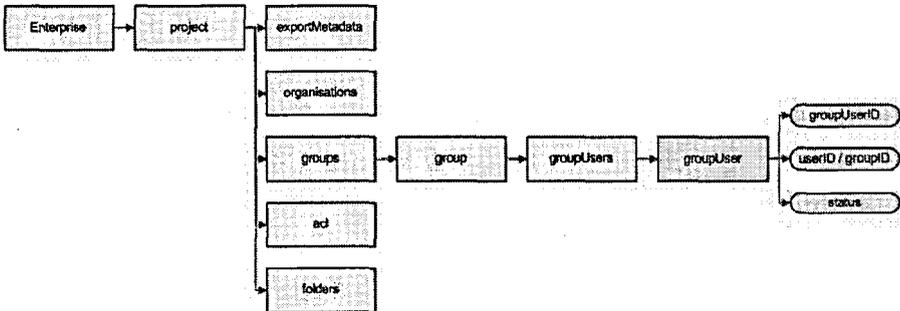
New Group object creation may contain groupUser object children; however all referenced user objects must have already been incrementally replicated in the destination system or be defined in the same incremental update as the group creation.

---

## 5.1.6 GroupUser Object Creation

---

The groupUser creation incremental update allows for new members to be added to groups which have already been replicated in the destination system. The xml definition of the groupUser must conform to that specified within the ncctp bulk data exchange standard, and must be held within an xml document that validates against the latest supported version of the ncctp incremental data exchange standard.

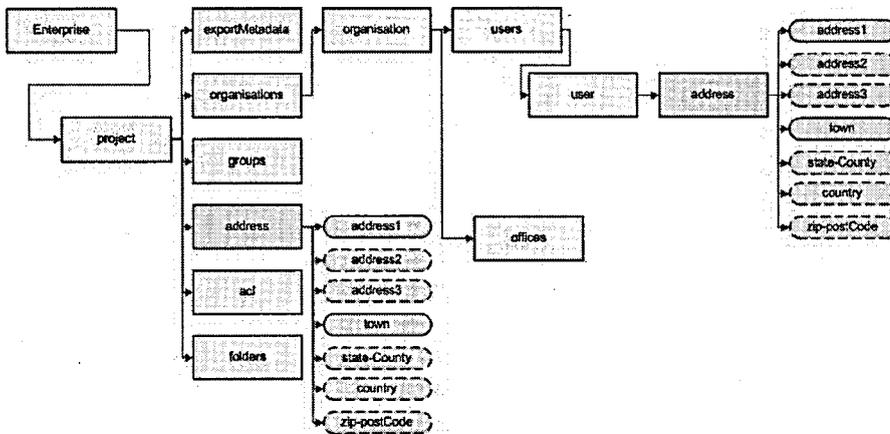


A groupUser will only be successfully added to a group if they are member of one of the organisations that are replicated between the source and destination system. Attempting to add a groupUser to a group where they already exist will be assumed to be a modification; however since groupUser information cannot be modified the request will be ignored.

Removal of groupUsers from a group is discussed in section 5.3

### 5.1.7 Address Object Creation

An address object can either be a child of a project, office or user object, and since not mandatory for project and user objects does not have to be included in those objects initial creation. The definition of an address passed must include the parent object type and its associated id property. All Address object definitions must comply with the latest supported version of the nctpt bulk data exchange standard.

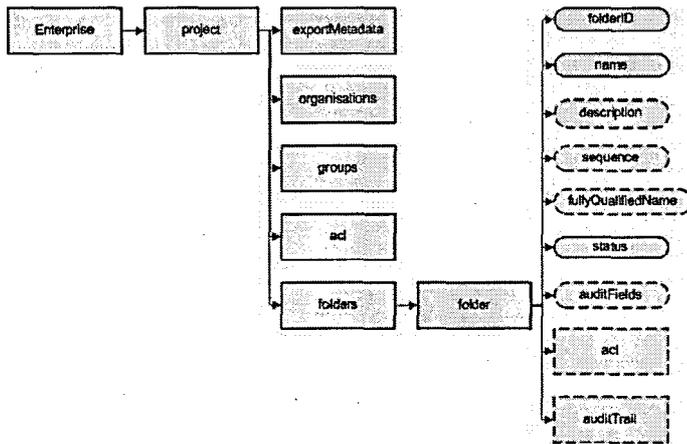


Incremental creation of address can only be applied to those objects that currently do not have any address data. Any Address Object specified for an object that already holds Address Data will be treated as a modification of the existing data, not the creation of a new Address object.

The create update instructs the destination system to add address data to the parent object specified in the update request. Changes to existing Address object data should be done using an address object update specified in section 5.2.

### 5.1.8 Folder Object Creation

A Folder object is a child of either the project object or another folder object, so in the create update both the type of parent object and the associated id property value must be specified. The definition of folder passed within the create update must comply with the latest supported version of the ncctp bulk data exchange standard, but should not contain any object children. Any child objects that the folder has must have been created after the folder was created so will be passed as additional create updates.

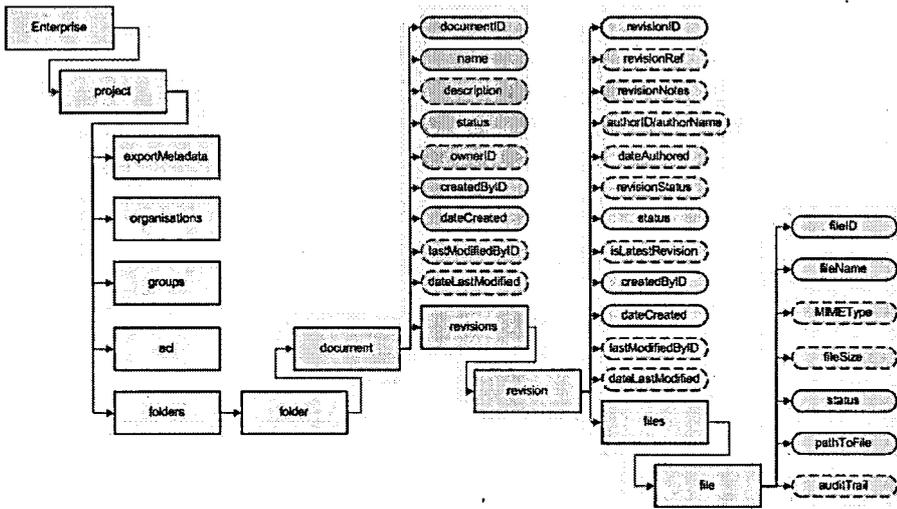


Included in the folder definition must be a single audit record that records when the object was created and by which user object, additional audit records should be passed as additional modify updates.

Should the source system support inheritance of access control lists, then the initial control list of the folder object should be included as part of the folder definition within the create update. All changes to this original access control list must be passed as modify updates to the destination system.

### 5.1.9 Document Object Creation

A Document object is a child of a folder object, so the create update must include the parent object type folder and its associated property value for id. The definition of the document passed must validate against the current supported version of the ncctp bulk data exchange standard with the exception been that it must not hold any revision children. All additional revision objects associated with the document must be created with separate create updates.



Included in the document definition must be a single audit record which shows when the object was created, the structure of the audit record must comply with the latest version of the ncctp bulk data exchange standard. All addition audit data for the document should be passed as separate modify updates.

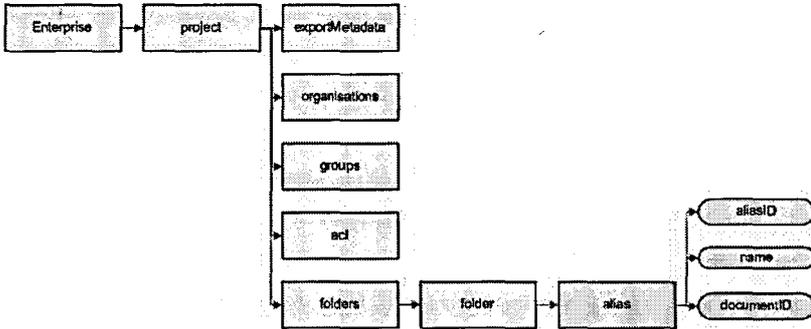
Should the source system support inheritance of access control list, then the list inherited from the parent object should be included as part of the document definition within the create update. All changes to this original access control list must be passed as modify updates to the destination system.

---

### 5.1.10 Alias Object Creation

---

An Alias object is a child of a folder object, so the create update must include the parent object type folder and its associated property value for id. The xml representation of the alias object passed must validate against the current supported version of the ncctp bulk data exchange standard.

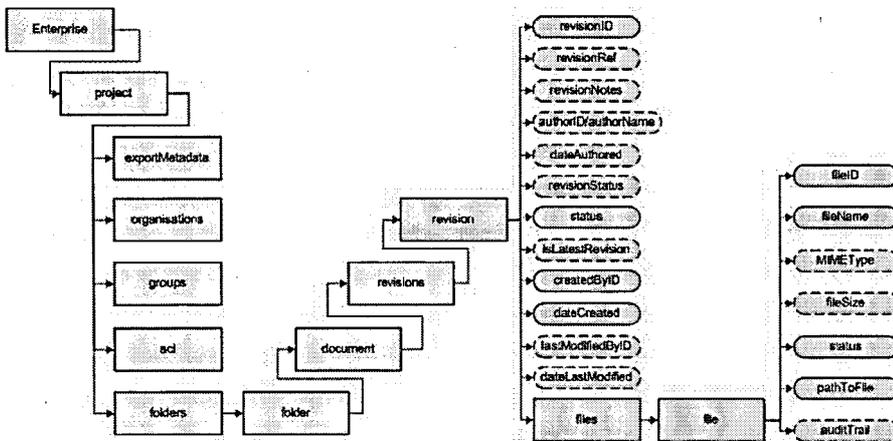


Included in the alias definition must be the minimum of a single audit record which shows when the object was created, the structure of this audit record must comply with the latest supported version of the ncctp bulk data exchange standard.

Any Object Referenced by the Alias Object must exist within the synchronised project and either has already been created in the destination system or is included in the same incremental update.

### 5.1.11 Revision Object Creation

A Revision object is the child of a document object, so the create update must include the parent document object and its id property to which the revision belongs. The definition of the revision object passed must comply with the latest supported version of the ncctp bulk data exchange standard.

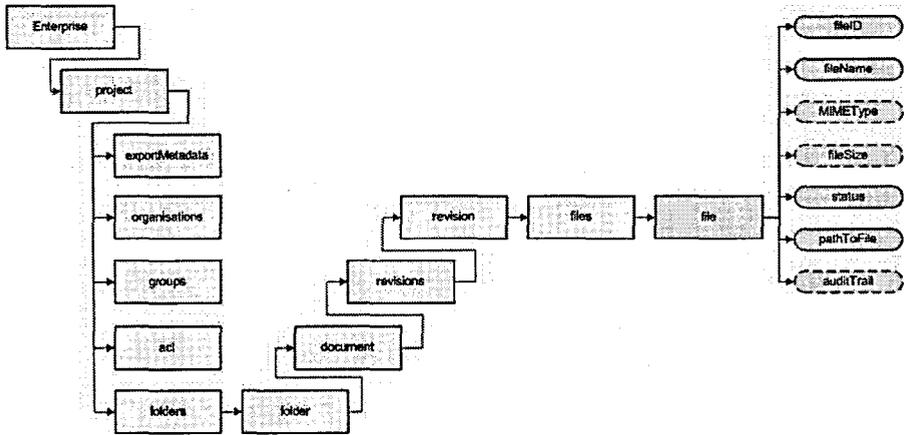


Accompanying the xml representation of the revision must be all the physical files that are referred to within the revision definition.

The Create Revision Object Incremental update instructs the destination system to add a revision to a document object which either already exist in the system or is included in the same incremental update. Any attempt to create a revision without its parent object being present should throw an error.

### 5.1.12 File Object Creation

A File object is the child of a revision object, so the create update must include the parent revision object and its associated id property. The definition of the file must comply with the latest supported version of the ncctp bulk data exchange standard.



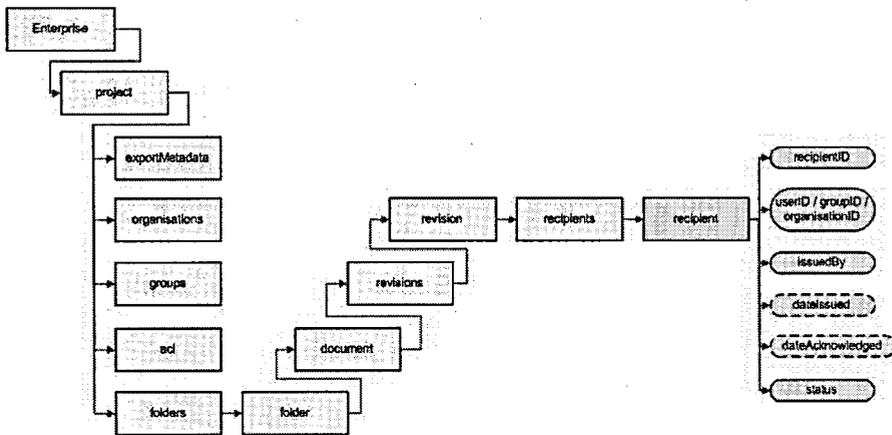
Accompanying the xml representation of the file must be the physical file which is referred to within the file definition.

The create file object Incremental update instructs the source system to add a file to a revision that either already exists in the system or is included in the same incremental update. Attempting to add a file to a revision that does not exist in the destination system should throw an error.

### 5.1.13 Recipient Object Creation

---

The Recipient Object Creation Incremental Update instructs the destination system to add a new recipient object to the revision specified in the request. The correct revision object is identified by the id property value and its position in the overall object tree. The xml definition of the recipient object must validate against the latest supported version of the nctcp bulk data exchange standard, and the entire document xml must comply with the nctcp incremental data exchange standard.

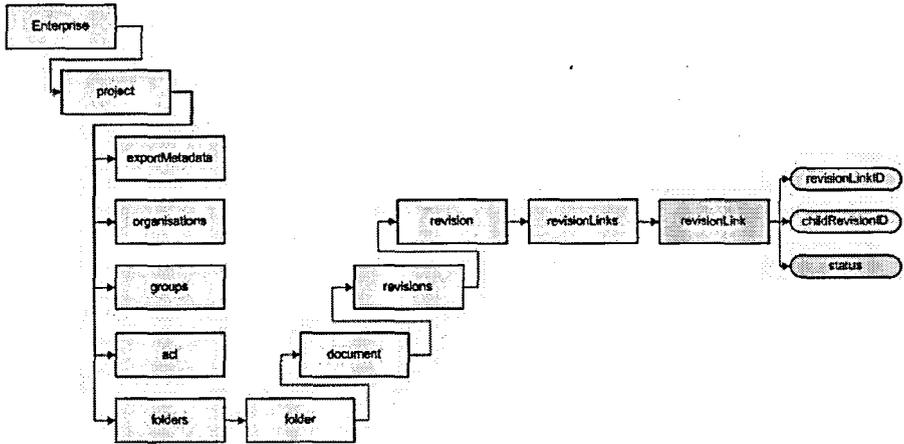


The property user specified in the recipient object must refer to a user object that has already been created in the destination system or is included in the same incremental update as the recipient object creation.

The id property value of the recipient must be unique inside the containing revision for a new recipient object to be added successfully. If a recipient with the same id property value already exists then the incremental update will be considered as an update of this existing recipient object.

### 5.1.14 RevisionLink Object Creation

The revisionLink Object Creation Update instructs the destination system to add a new revisionLink object to the revision object specified in the request. The correct revision is identified by the revision id property value and the position of the revision object in the object tree. The overall xml document sent containing the revisionLink Creation update must validate against the latest supported version of the nctp incremental data exchange standard; while the xml definition of the revisionLink object must conform to the object specification defined in the nctp bulk data exchange standard.



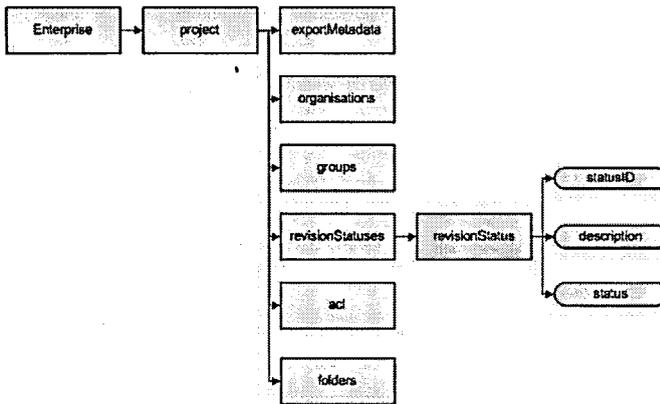
The revision object referenced in the RevisionLink must already have been created in the destination system or be included in the same incremental update as the revisionLink object itself.

The id property of the revision link object must be unique amongst all other revisionLink objects for that revision. If a revisionLink already exists in the destination within the specified revision that has the same id property value then the create update will be assumed to be a modification, adjusting the values of the pre-existing object.

### 5.1.15 RevisionStatus Object Creation

---

The revisionStatus Object update instructs the destination to add a revision status to those statuses which document revisions can be assigned. The id property of the new revisionStatus must be unique amongst all those already existing on the project else this create update will be assumed an update of an existing revision status object. The xml definition of the new revisionStatus object must comply with the latest supported version of the ncctb bulk data exchange standard; while the xml document in which the new revision is contained must validate against the ncctb incremental data exchange standard.

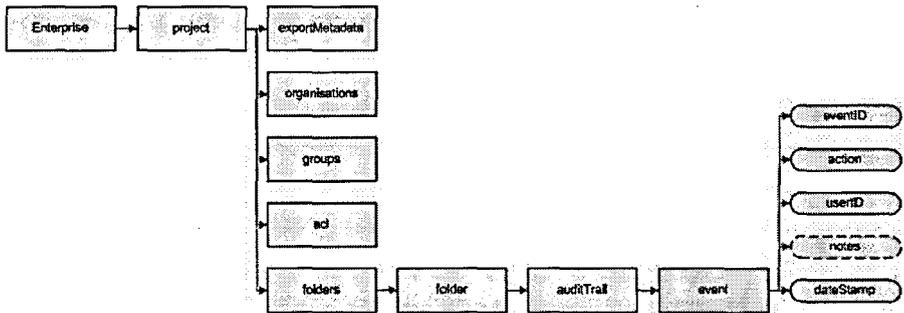


New revisionStatus objects that had an initial status property set to Deleted should not be assumed by the source system to have been created in the destination; due to the different ways in which systems handle the deletion of these types of object. It is recommended that objects always be created with a status property value of active to ensure maximum compatibility.

### 5.1.16 AuditTrail Object Creation

Audit Trail Objects hold information about the history of an object from its creation, user interaction then potentially its deletion. All auditable interaction with an object in the source system must be replicated in the destination system, which is done through the creation of additional auditTrail objects. Unlike other object types auditTrail objects can only be created, they can never be modified or deleted.

The incremental xml update document to create a new auditTrail object must contain an audit object xml definition that conforms to the nctp bulk data exchange standard; while the entire xml document must validate against the nctp incremental data exchange standard.



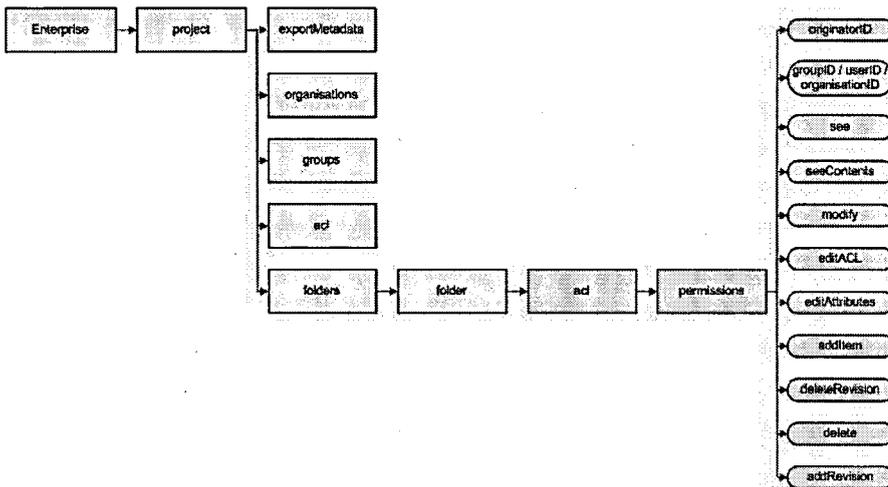
The object to append the auditTrail event is defined by the position of the object within the object tree structure. Any user, group or organisation referenced in the auditTrail event must have already been created in the destination system or be included in the same incremental update as this auditTrail Object.

### 5.1.17 Access Control List Object Creation

---

An Access Control List object holds the rights users, group and organisation have on other objects in the collaborative system. Any application of permissions to an object should be incrementally updated in the destination system to ensure that object security is maintained. Therefore the creation of an access control list object, only refers to the initial application of permissions to an object that had no permissions applied to it before. Changes to Access Control Lists should be maintained through the modification update.

Initial XML definitions of access control list should to the ncctp bulk data exchange standard while the XML incremental update document should validate against the ncctp incremental data exchange standard.



The object to which the new access control list object is to be applied is defined by the position of the object within the object tree structure.

Depending upon the purpose of the incremental transfer access control list object creations may be ignored as all objects will be set to read only in the destination system.

---

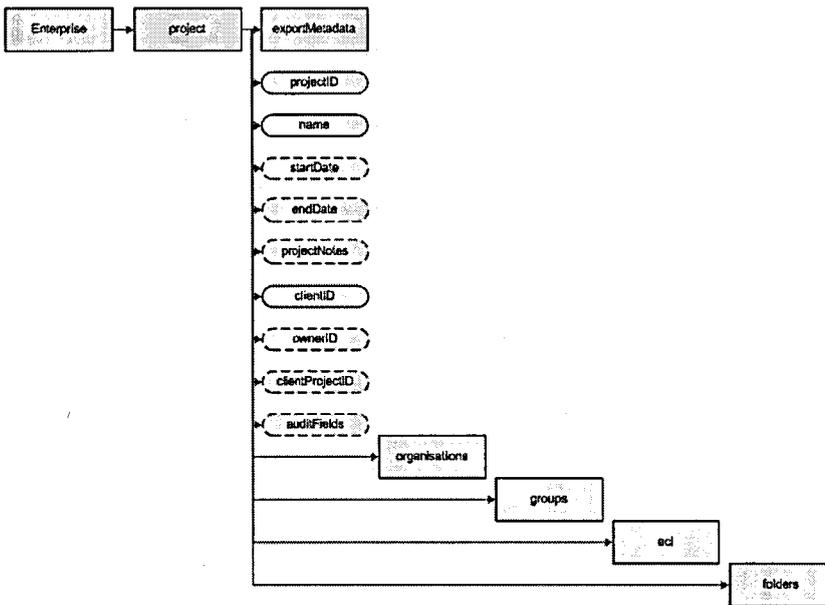
## **5.2 Object Modifications**

---

Any changes to incrementally replicated objects within the source system must be made to objects in the destination system in order to keep the two systems synchronised. Modifications are categorised as changes to existing objects metadata only. For Example adjusting the name of a folder would be classified as a modification while adding a new member to a group would be a creation.

### 5.2.1 Project Object Modification

Changes in the source system to the project objects metadata, audit history or access control lists needs to be replicated in the destination system, however the addition of new child object folders, organisation etc. do not constitute a modification of the project object itself. Project Object updates definitions should conform to the latest supported version of the ncctp incremental data exchange standard.



Only metadata changes to the project object will be discussed in this section, audit History changes and access control list changes have their own section as they are universally applicable.

Project object updates will only include properties that have changed, with the exception of those required by the schema for object identification. For Example the presence of an endDate element in the update will mean that the endDate property of the project object has changed in the source system. While its absence for the xml definition of the project object will indicate that the property value is unchanged.

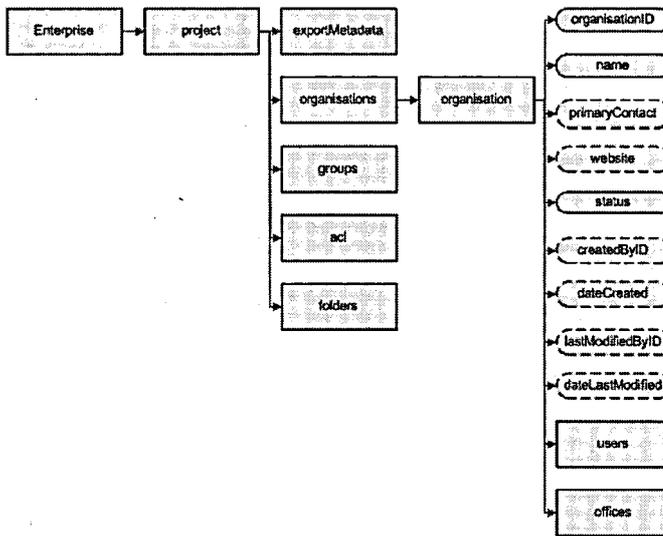
---

The project id property cannot be changed through the Project Object Modification process.

### 5.2.2 Organisation Object Modification

---

The Organisation object modification update will allow for changes in an organisation objects properties on the source system to be duplicated in the destination system. Only changes to Organisation objects properties should be done through this modification, additional office or user objects do not constitute an update to the organisation object. All Organisation Object Modifications must validate against the latest version of the nccpt incremental data exchange standard.



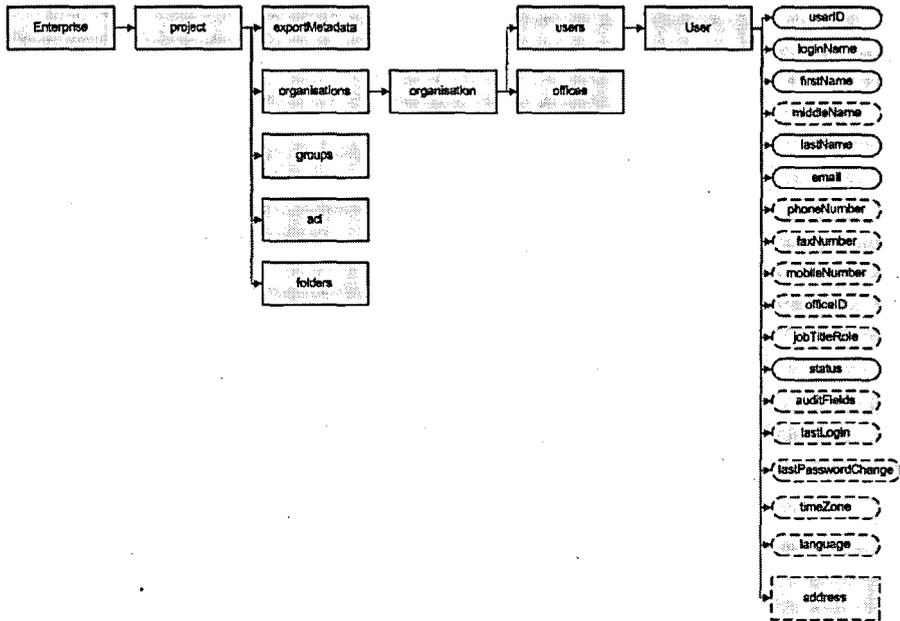
Organisation object modification updates will only include properties that have been changed in the source system, with the exception of those which are required by the schema to identify the object. The absence of an element from the xml definition will indicate that the object property value has not changed in the source system.

This Update instructs the destination system to change object property values for the organisation specified in the request.

The Organisation objects id property cannot be adjusted using this method.

### 5.2.3 User Object Modification

The user object modification update allows for properties of the user object that have been changed in the source system to be replicated in the destination system. Any changes to child objects of the user object are covered in additional sections, and do not constitute an update of the user object itself. Updates to the user object must comply with the latest supported version of the nccpt incremental data exchange standard.



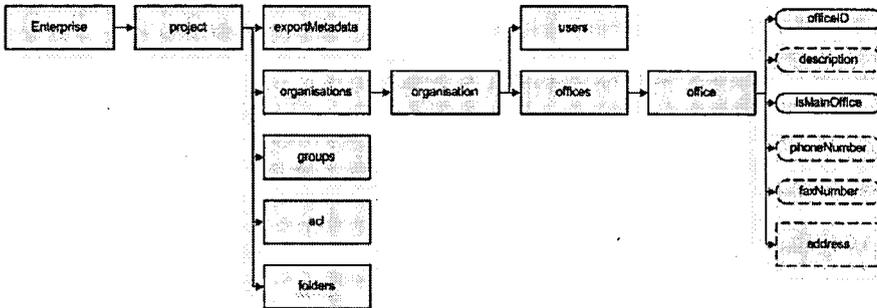
User object modification updates will only include properties that have changed in the source system, any property not included in the xml definition of the user should be assumed to be unchanged. The only exception to this is for those properties that are required for object identification by the destination system.

This update instructs the destination system to change object properties for the user object specified, with the exception of the id property that cannot be modified using this method.

### 5.2.4 Office Object Modification

---

The office object modification allows for the properties of an office object that have been changed in the source system to be updated in the destination system. Any changes to child objects, namely address objects, are covered in additional sections and do not constitute a change to the office object itself. Updates to office objects must comply with the latest supported version of the nctpt incremental data exchange standard.

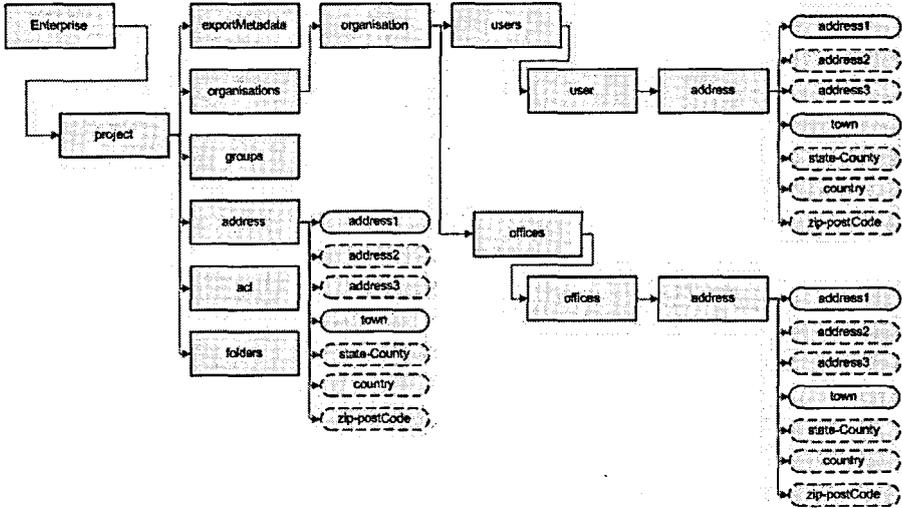


With the exception of those object properties that are required for object identification only those properties that have changed in the source system should be included in the xml representation of the office object. Any property not included in the definition of the office object will not have changed value in the source system.

This update gives the destination system all the information required to adjust the office object so it is synchronised with the office that exists in the source system. All properties with the exception of the office id property can be modified using this method.

## 5.2.5 Address Object Modification

The Address object modification allows for properties of an address object to be modified in the destination system. Any updates to the address object properties must be valid to the latest supported version of the nctpt incremental data exchange standard.

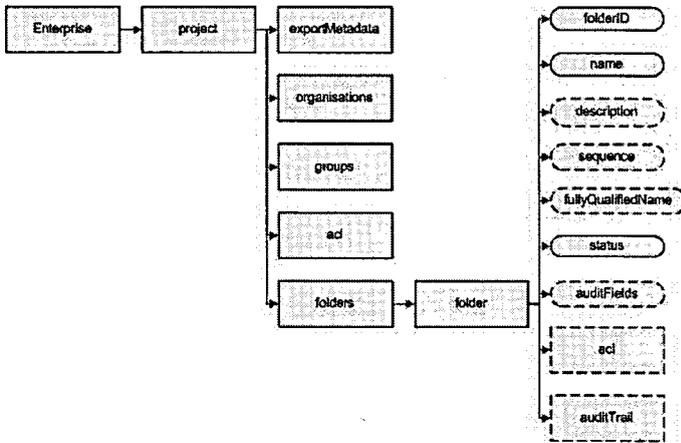


Address object modification updates should include reference to the parent object as a means of identification. All address object properties can be modified using this method, and those included in the xml definition, with the exception of those required by the schema will be assumed to have changed.

### 5.2.6 Folder Object Modification

---

Changes to the properties of folder object in the source system are replicated in the destination system by the folder object modification update. Any changes to child objects, documents, alias' or additional folder are discussed in additional sections along with audit and access control list updates. Modification updates must conform to the latest supported version of the nctcp incremental data exchange standard.

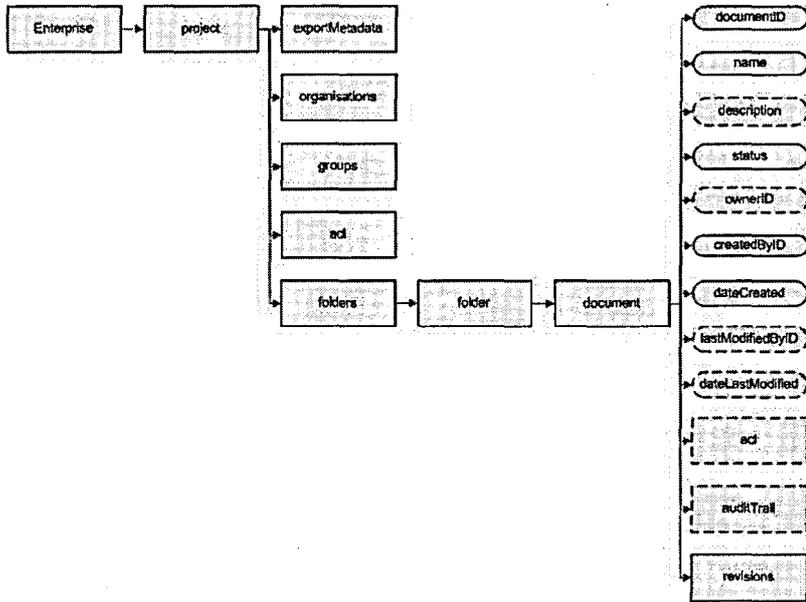


Folder object modification updates should only include property values that have changed in the destination system, along with those required by the schema for object identification. All properties not included can be assumed not to have changed.

All folder property values with the exception of the id property can be modified using the method.

## 5.2.7 Document Object Modification

The Document object modification update allows for changes to a document objects property values in the source system to be replicated in the destination system. Any changes to child objects, namely revisions, audit and access control lists will be discussed in additional sections and do not constitute a modification of the document object itself. All document object modification updates must valid against the latest supported version of the ncctp incremental data exchange standard.



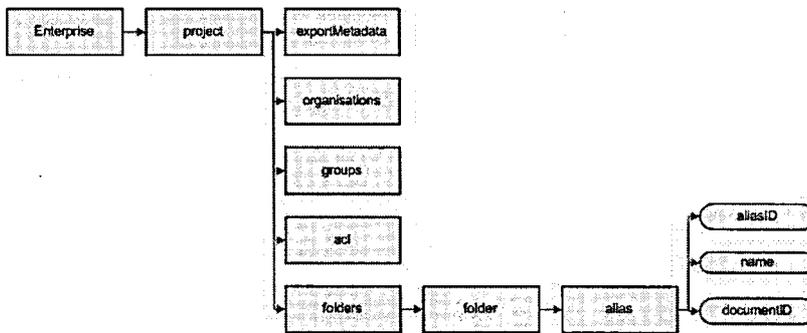
Document object modification updates should only include those property values that have changed in the source system along with those required by the schema to identify the object. All object property values not included in the update should be assumed not to have changed in the source system.

All document property values can be modified using this method with the exception of the id property value required for object identification.

## 5.2.8 Alias Object Modification

---

Property changes to an alias object in the source system are replicated in the destination system using the alias object modification update. Changes to alias child objects like access control lists and audit will be discussed in additional sections, and do not constitute modifications to the alias object itself. Alias modification updates must be valid against the latest supported version of the nccpt incremental data exchange standard.

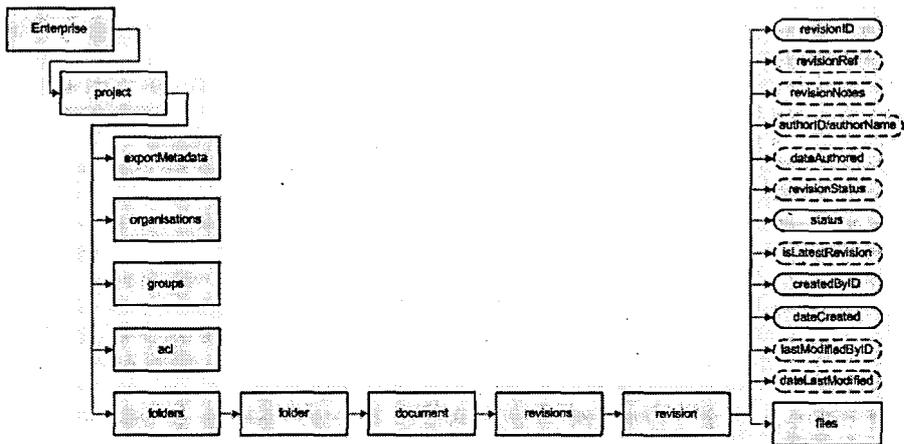


The xml definition of alias passed in the incremental update will only include those object properties that have changed in the source system and those required to identify the alias object, namely the id property value. All those property values not present in the xml definition can be assumed not to have changed in the source system.

All alias object properties with the exception of the id property can be modified using this method. However if the property documented is modified the newly referenced document must either already exist in the destination system or be included in the same incremental update.

## 5.2.9 Revision Object Modification

The revision object modification update allows for changes made in the source system to a revision object to be made in the destination system to the same revision object. Any changes to revision child objects do not constitute a change to the revision object and only modification to revision object properties are covered in this method. All Revision object modification updates must conform to the latest supported version of the ncctp incremental data exchange standard.

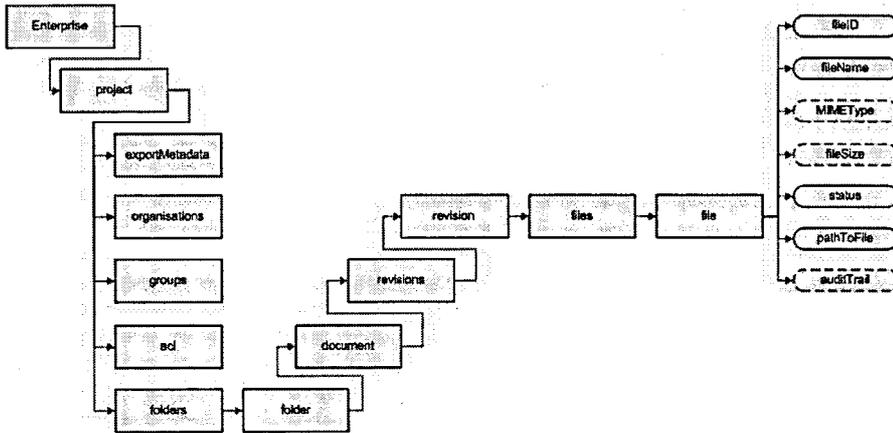


Revision Object modification updates only should include property values that have changed in the source system along with those properties required by the schema for identification. All property values not included in the update are assumed not to have changed in the source system.

All Revision properties with the exception of the id property can be changed using this method. However any changes in the revisionStatus property must be to one of the values specified for the project. If the change is to a value not already duplicated in the source system then this additional revision status must be included in the same incremental update.

### 5.2.10 File Object Modification

The file object modification update is used to replicate incremental changes to the file object made in the source system, to the file object in the destination system. Any changes to file object children do not constitute changes to the file object and only those made to properties should be updated using this method. All file modifications incremental updates created must conform to the latest supported version of the ncctp incremental data exchange standard.



File Object Modification updates should only include those properties that have changed in the destination system, with the exception of those required for file object identification. All object properties not included in the update should be assumed to have remained unchanged.

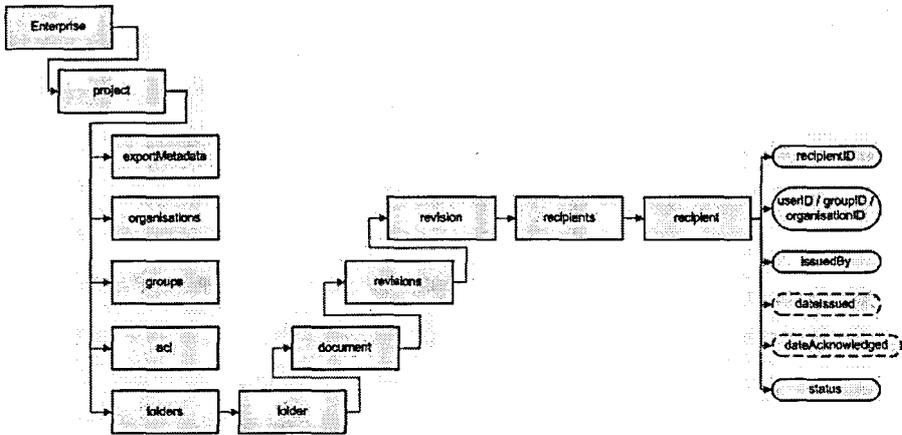
All file object property values with the exception of the id property value can be updated in the destination system using this method.

---

### 5.2.11 Recipient Object Modification

---

The recipient object modification update is used to replicate changes in the destination system that have occurred to the recipient object in the source system, for example the dateAcknowledged property has now been set. All updates to recipient object must validate against the ncttp incremental updating specification. The recipient object to be updated is identified by the value of the recipientID property included in the recipient update request and its position in the object tree.



Recipient Object modifications should only include those property values that have changed and those which are required by the schema. Any property value not included in the update can be assumed by the importing system to not have changed.

All recipient object properties with the exception of recipientID can be updated by this method; however change to an attribute other than dateAcknowledge should be made with the deletion of the existing recipient object and the creation of a new one.

### **5.2.12 Access Control List Object Modification**

---

An Access Control List Modification can either take the form of adjusting the permissions assigned to another object, creating permissions for a new object or removing permissions. Any incremental update of an object's access control list should include all permissions specified on the object, regardless of whether or not they have changed.

When the destination system is processing an incremental access control list update, it will interpret the contents as follows:

If a permission object exists for another object in the destination system, but is not included in the incremental update then this permission object has been removed in the source system.

If a permission object does not exist in the destination system, but is included in the incremental access control list object update then this permission object has been added to the source system and therefore should be added to the destination system.

If a permission object exists for an object in both the destination system and in the incremental update, then the destination system knows that either this permission object has been modified or is unchanged. The destination system will therefore need to compare the rights granted in the in both instances and update the destination system if required.

The object to which the modified access control list belongs is denoted by its position in the overall object tree. All incremental access control list modifications must validate against the latest supported version of the ncctp incremental data exchange standard.

---

### **5.3 Object Deletion**

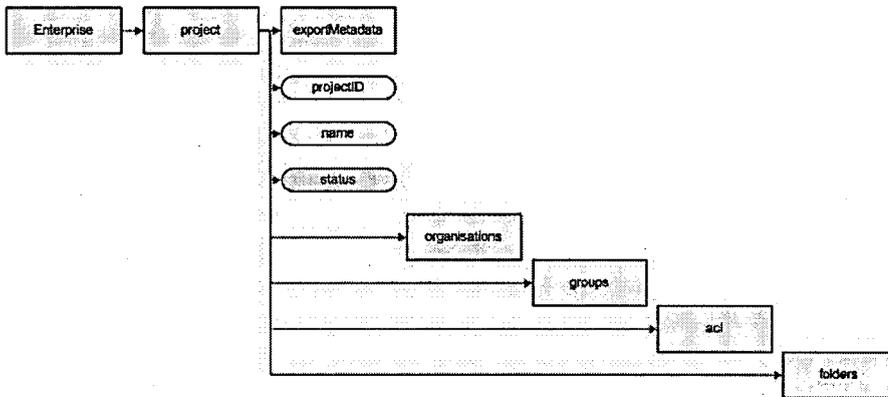
---

All new objects deleted in the source system must be included in any incremental project update as deleted objects. Deletion is indicated to the destination system by the setting of the objects status property value to 'Deleted'.

### 5.3.1 Project Object Deletion

---

A Project object deletion incremental update instructs the destination that the project object identified by its projectID property should be marked as deleted. The requested deletion of the project object is indicated by the setting of the status property value to 'Deleted'. All project deletion incremental updates must validate against the latest supported version of the ncctp incremental data exchange standard.



The deletion of a project object will lead to the deletion of all subsequent child objects contained in the project; therefore it is not required to set the statuses for all these objects in the incremental update.

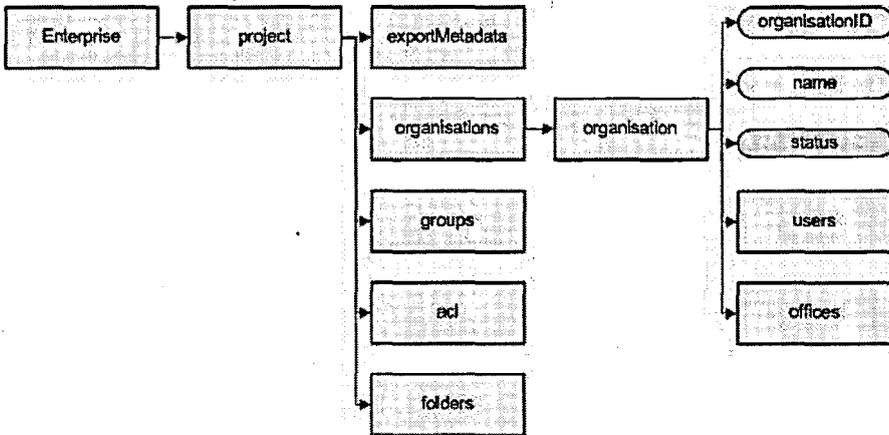
Although it is possible to delete a project object through an incremental update it is recommended that the functionality is not implemented, thus reducing the risk of accidental deletion.

---

### 5.3.2 Organisation Object Deletion

---

This incremental update instructs the destination system to remove an organisation object, and it triggered by the organisation object being deleted in the source system. The deletion of the organisation object is specified in the incremental update by setting the property object to 'Deleted'. All Organisation object deletions must validate against the latest supported version of the ncctps incremental data exchange standard.



The deletion of Organisation object from the system will automaticall delete all user and office objects that it contains, therefore it is not required to set the status property value for these objects individually.

Note that the deletion of an organisation object will not remove any references that exist to this organisation in oother oibjects in the system. For example no access control lists that contain rights for the organisation will be removed.

If the destination system supported the retention of deleted organisation objects then any changes to additional property values for the object should be make in addition to setting the status propewrty value to 'Deleted'.

### **5.3.3 User Object Deletion**

---

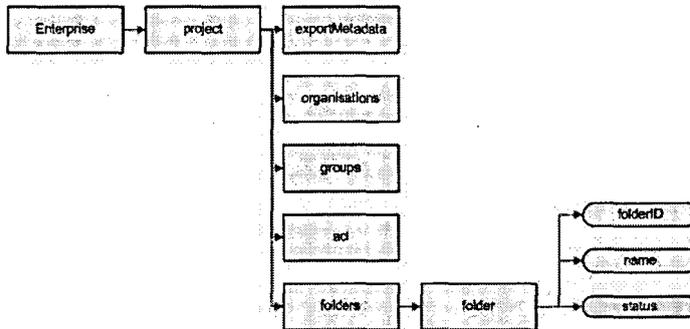
A Folder object deletion incremental update instructs the destination that the folder object identified by its folderID prop

---

### 5.3.6 Folder Object Deletion

---

A Folder object deletion incremental update instructs the destination that the folder object identified by its folderID property and its position in the object tree should be marked as deleted. The requested deletion of the folder object is indicated by the setting of the status properties value to Deleted. All folder deletion incremental updates must be valid against the latest supported version of the nctcp incremental data exchange standard.



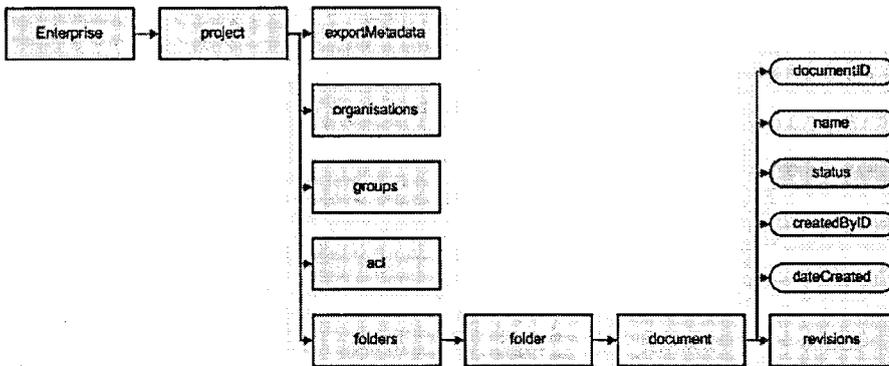
The deletion of a folder object will lead to the cascading deletion of all child objects contained under the folder in this branch of the object tree. Therefore it will not be required to set the status of child objects to Deleted within the incremental update.

The deletion of folder child objects such as documents and revisions will not affect any objects that reference these objects.

### 5.3.7 Document Object Deletion

---

The Document Deletion incremental update allows for document object selected in the source system to also be deleted in the destination system. The deletion of the document object is indicated by setting its status property to Deleted, in a document update that is valid against the latest supported version of the ncctp incremental data exchange standard. The document object to be deleted by the destination system is identified by the documented object property and the document objects position in the ncctp object tree.

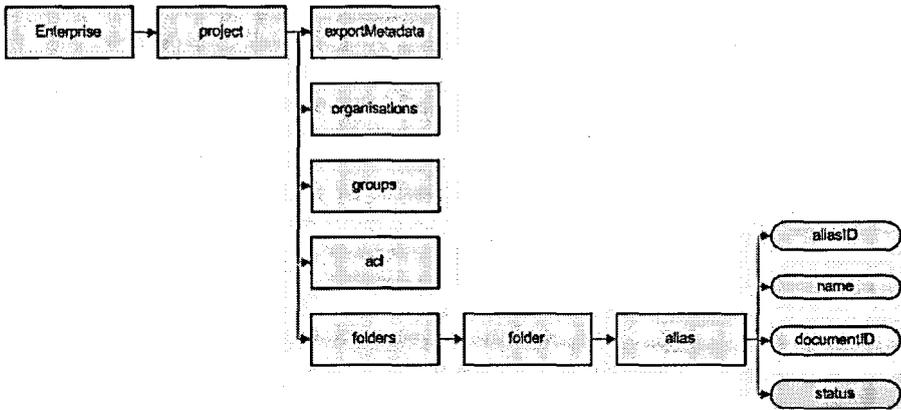


The Deletion of a Document object will lead to the deletion of all revision objects it contains, and subsequent objects below it in the object tree. It is not required to set the status of these child objects to Deleted in an incremental update.

The deletion of a document object will not affect any alias objects that reference this document, or any objects that reference revisions contained within the document.

### 5.3.8 Alias Object Deletion

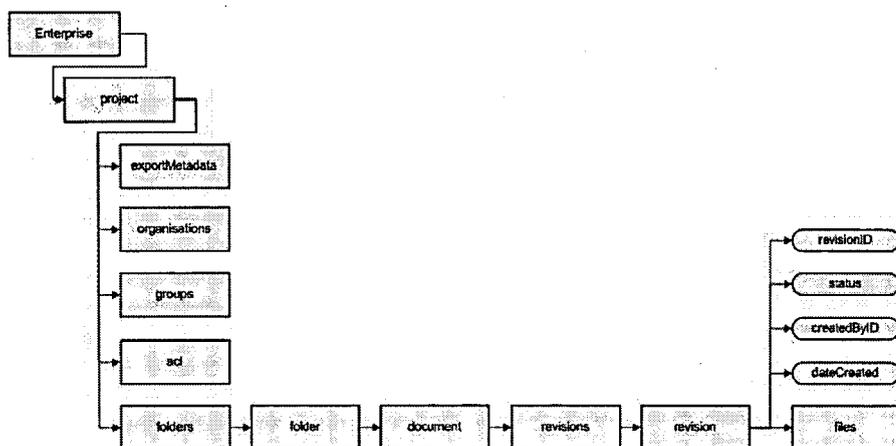
Deletion of Alias objects in the source system is replicated in the destination system through the alias object deletion incremental update. The alias update passed in the deletion request must conform to the latest support version of the ncctp incremental data exchange standard. Alias object deletion is achieved by setting the alias objects status property to Deleted. The alias which is to be marked as deleted in the destination system is identified by the aliased property and the object tree in which the alias object is contained.



Deletion of an alias object will not delete the object to which the alias refers.

### 5.3.9 Revision Object Deletion

The revision object deletion update allows for the deletion of revision objects in the destination system. The definition of revision passed to the destination system in the incremental update must confirm to the latest supported version of the nctp incremental data exchange standard. Revision Object deletion is achieved by setting the status property of the revision to Deleted. The destination system identifies which revision is to be deleted by the id property of the revision and the tree structure in which the revision is held.



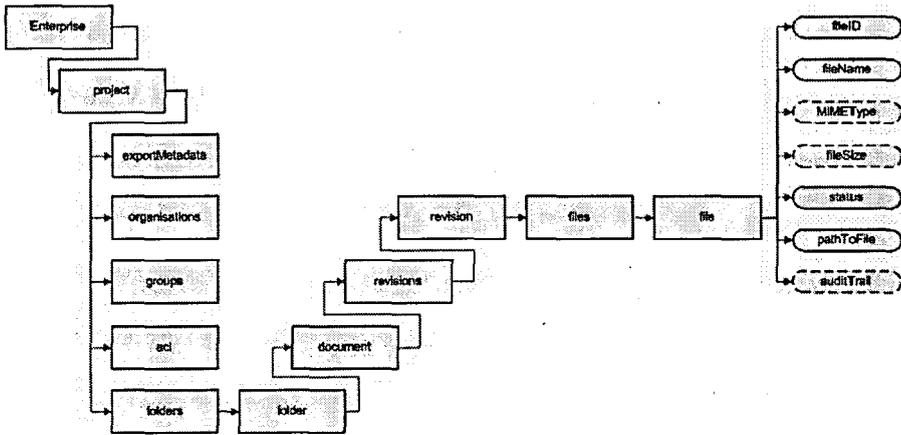
Deletion of a revision object will lead to the deletion of all child objects held within that revision. It will not be required to set the status of all the child objects individually to Deleted.

A Revision Object can only be deleted if the document object holds more than one revision.

The revision object deletion will not affect any revision links that point to this revision.

### 5.3.10 File Object Deletion

The file object deletion update allows for the deletion of file objects in the destination system. The definition of file passed to the destination system in the incremental update must confirm to the latest supported version of the ncftp incremental data exchange standard. File Object deletion is achieved by setting the status property of the file object to Deleted. The destination system identifies which file is to be deleted by the id property of the file and the position within the object tree the file is positioned.

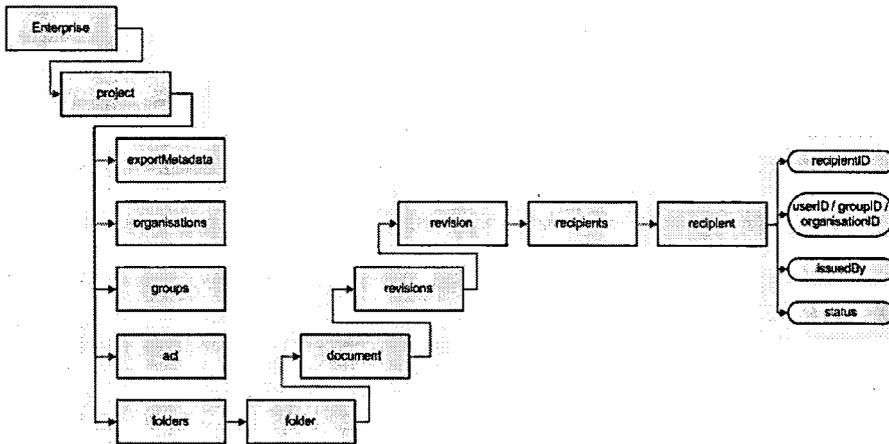


A File object can only be deleted if the revision to which it belongs holds more the one file.

### 5.3.11 Recipient Object Deletion

The recipient object deletion incremental update allows for recipient objects deleted in the source system to be deleted in the destination system. The deletion of the object is indicated by setting the status property value of the recipient object to Deleted. If the destination system keeps deleted objects then another changes to the recipient object included in the same incremental update must also be made to the object; however if the object is deleted from the destination system then these changes do not need to be made.

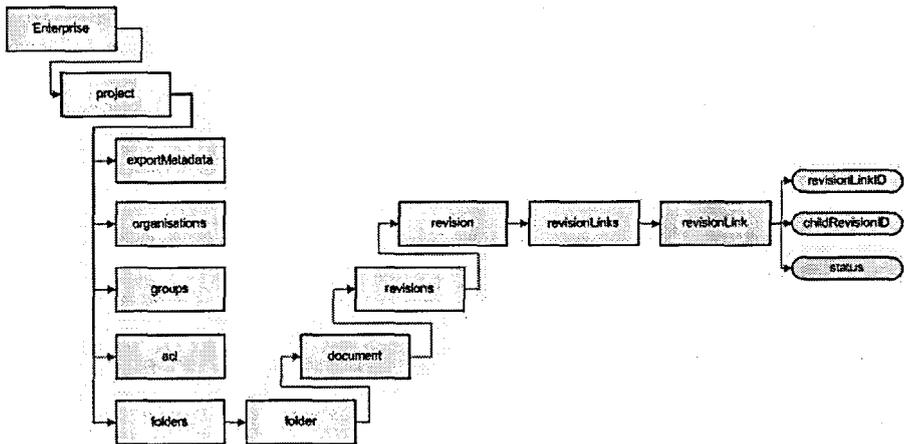
All incremental updates that delete recipient objects must validate against the latest ncctp incremental data exchange standard.



It is not advised that recipient objects are deleted as they hold audit information about who had seen the information, when they were issued it and the date of acknowledgement that may not be maintained in the general object audit trail.

### 5.3.12 RevisionLink Object Deletion

RevisionLink Object deletion allows linkages between revisions to be removed from the revision objects in the destination system; allowing it to represent the revision object in the source system. The destination is informed that the link has been deleted by the setting of the status property value in the revisionLink object to Deleted. If the destination system continues to hold revisionLinks after they are deleted then another changes, represented by adjustments to other property values should also be made. Incremental revisionLink Delete Updates must validate against the nctcp incremental data exchange specification.



## **5.4 Object Translation**

---

Object translations are the movement of an object from one position in the object tree to another position in the object tree; however objects can only be moved to a position in the tree that is supported by the ncctp specification. For Example it would not be possible to move a document object from being a child of a folder object to being the child of an organisation object because this parent child relationship is not supported by the specification. Object translations represent a change in the object tree structure, with common examples being a user changing organisation, or a folder being moved in the system.

The following two sections show how object translations for both the folder and organisation branches are represented in the ncctp incremental data exchange specification.

---

### **5.4.1 Organisation Branch Translations**

---

Object translations within the organisation branch can only take the form of user objects being moved from one organisation object to another organisation object. Since the user object properties have already been defined in the destination system all it is required from the update is information that indicates that the user object has moved organisation. This is achieved by placing the user object as a child of its new organisation in the incremental update. The destination system then recognises the user by its userID property value and the fact that it is in a different organisation, to the one in which it currently exists.

Any changes to the user objects property values, such as a change of office reference can also be included in the same incremental update.

Note if a user object has been created and then moved between two incremental updates then the full definition of the user object must be passed in the incremental update.

### **5.4.2 Folder Branch Translations**

---

Object translations in the folder branch can take the form of folder, document or alias objects moving their position in the overall object tree. Since these objects have already been created in the destination system it is not required to pass their full definitions in any incremental update. As object id property values are unique amongst their object class, objects can be identified from this information alone.

The translation of objects is achieved by placing the translated object into its new position in the object tree within the incremental update. When importing the incremental update the destination system will recognise the object and the fact that it is now located in a different place in the object tree.

Any changes to properties of the moved object since the last incremental update can also be passed along with the object definition.

Note if an object was created and then moved between two incremental updates or between the bulk exchange and first incremental update then the full object definition must be passed in the incremental update.

---

**6.1 Object Creation**

---

The following example xml documents illustrate how various ncctp objects are created within an incremental update. These files are for illustrate purposes only as likely incremental updates will contain a number of creation, modification, deletion and translations actions.

### 6.1.1 Organisation Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new organisation object (organ1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations>
      <organisation organisationID="organ1">
        <name>Organisation Alpha</name>
        <primaryContact>scott moses</primaryContact>
        <status>Active</status>
        <createdByID>adminUser</createdByID>
        <dateCreated>2006-04-01T21:54:00</dateCreated>
        <lastModifiedByID>adminUser</lastModifiedByID>
        <dateLastModified>2006-04-01T21:54:00</dateLastModified>
      <users>
        <user userID="user4">
          <loginName>testUser4</loginName>
          <firstName>Test</firstName>
        </user>
      </users>
    </organisation>
  </project>
</Enterprise>
```

```

<middleName>4</middleName>
<lastName>User</lastName>
<email>test.user4@ncctp.net</email>
<phoneNumber>01509805000</phoneNumber>
<faxNumber>01509805001</faxNumber>
<mobileNumber>88888888888</mobileNumber>
<officeID>office1</officeID>
<jobTitleRole>Engineer</jobTitleRole>
<status>Active</status>
<createdByID>adminUser</createdByID>
<dateCreated>2006-04-01T21:54:00</dateCreated>
<lastModifiedByID>adminUser</lastModifiedByID>
<dateLastModified>2006-04-
01T21:54:00</dateLastModified>
<lastLogin>2006-04-01T21:54:00</lastLogin>
<lastPasswordChange>2006-04-
01T21:54:00</lastPasswordChange>
<timeZone></timeZone>
<language></language>
<address>
  <address1>address1</address1>
  <address2>address2</address2>
  <address3>address3</address3>
  <town>town</town>
  <state-County>county</state-County>
  <country>country</country>
  <zip-postCode>postcode</zip-postCode>
</address>
</user>
</users>
</offices>
<office officeID="office1">
  <description>Main Office</description>
  <isMainOffice>true</isMainOffice>
  <phoneNumber>88888888888</phoneNumber>
  <faxNumber>88888888888</faxNumber>
  <address>
    <address1>address1</address1>
    <address2>address2</address2>
    <address3>address3</address3>
    <town>town</town>
    <state-County>county</state-County>
    <country>country</country>
    <zip-postCode>postcode</zip-postCode>
  </address>
</office>
</offices>
</organisation>
</organisations>
<groups/>
<acl/>
<folders/>
</project>
</Enterprise>

```

## 6.1.2 User Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new user object (user5), within an organisation object (organ1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations>
      <organisation organisationID="organ1">
        <name>Organisation Alpha</name>
        <status>Active</status>
        <users>
          <user userID="user5">
            <loginName>testUser5</loginName>
            <firstName>Test</firstName>
            <middleName>5</middleName>
            <lastName>User</lastName>
            <email>test.user5@ncctp.net</email>
            <phoneNumber>01509805000</phoneNumber>
          </user>
        </users>
      </organisation>
    </organisations>
  </project>
</Enterprise>
```

01T21:54:00</dateLastModified>

01T21:54:00</lastPasswordChange>

<faxNumber>01509805001</faxNumber>  
<mobileNumber>888888888888</mobileNumber>  
<officeID>office1</officeID>  
<jobTitleRole>Engineer</jobTitleRole>  
<status>Active</status>  
<createdByID>adminUser</createdByID>  
<dateCreated>2006-04-01T21:54:00</dateCreated>  
<lastModifiedByID>adminUser</lastModifiedByID>  
<dateLastModified>2006-04-

<lastLogin>2006-04-01T21:54:00</lastLogin>  
<lastPasswordChange>2006-04-

<timeZone></timeZone>  
<language></language>  
<address>  
    <address1>address1</address1>  
    <address2>address2</address2>  
    <address3>address3</address3>  
    <town>town</town>  
    <state-County>county</state-County>  
    <country>country</country>  
    <zip-postCode>postcode</zip-postCode>  
</address>

</user>

</users>

<offices/>

</organisation>

</organisations>

<groups/>

<ad/>

<folders/>

</project>

</Enterprise>

### 6.1.3 Office Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new office object (office2), within an organisation object (organ1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations>
      <organisation organisationID="organ1">
        <name>Organisation Alpha</name>
        <status>Active</status>
        <users/>
        <offices>
          <office officeID="office2">
            <description>Northern Office</description>
            <isMainOffice>>false</isMainOffice>
            <phoneNumber>8888888888</phoneNumber>
            <faxNumber>8888888888</faxNumber>
            <address>
```

---

```
<address1>address1</address1>
<address2>address2</address2>
<address3>address3</address3>
<town>town</town>
<state-County>county</state-County>
<country>country</country>
<zip-postCode>postcode</zip-postCode>
</address>
</office>
</offices>
</organisation>
</organisations>
<groups/>
<acl/>
<folders/>
</project>
</Enterprise>
```

## 6.1.4 Group Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new group object (group1), within the project object (proj12345).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups>
      <group groupID="group1">
        <name>New Group</name>
        <status>Active</status>
        <groupUsers/>
      </group>
    </groups>
    <acl/>
    <folders/>
  </project>
</Enterprise>
```

---

## 6.1.5 GroupUser Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new groupUser object (group1user2), within the group object (group1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups>
      <group groupID="group1">
        <name>New Group Renamed</name>
        <status>Active</status>
        <groupUsers>
          <groupUser groupUserID="group1user2">
            <userID>user2</userID>
            <status>Active</status>
          </groupUser>
        </groupUsers>
      </group>
    </groups>
  </project>
</Enterprise>
```

## Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

---

```
    </groups>  
    <acl/>  
    <folders/>  
  </project>  
</Enterprise>
```

---

## 6.1.6 Address Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new address object, within the project object (proj12345).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <address>
      <address1>address1 Modified</address1>
      <address2>address2 Modified</address1>
      <address3>address3 Modified</address1>
      <town>town Modified</town>
      <state-County>changed</state-County>
      <country>changed</country>
      <zip-postCode>HP109PN</zip-postCode>
    </address>
  </project>
</Enterprise>
```

```
        <folders/>
    </project>
</Enterprise>
```

b) The following example xml file shows the incremental update required for the creation of a new address object, within the user object (xxxxx).

---

## 6.1.7 Folder Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new folder object (folderL1N1), within the project object (proj12345).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Drawings</name>
        <description>This Folder Will Hold all Project Drawings</description>
        <sequence>1</sequence>
        <fullQualifiedName>Drawings</fullQualifiedName>
        <status>Active</status>
        <createdByID>AdminUser</createdByID>
        <dateCreated>2006-06-12T14:34:20</dateCreated>
      </folder>
    </folders>
  </project>
</Enterprise>
```

```
<lastModifiedByID>AdminUser</lastModifiedByID>
<dateLastModified>2006-06-12T14:34:20</dateLastModified>
<acl>
  <permissions originatorID="adminUser">
    <groupID>organ1</groupID>
    <see>true</see>
    <seeContents>true</seeContents>
    <modify>true</modify>
    <editACL>true</editACL>
    <editAttributes>true</editAttributes>
    <addItem>true</addItem>
    <deleteRevision>true</deleteRevision>
    <delete>true</delete>
    <addRevision>true</addRevision>
  </permissions>
</acl>
<auditTrail>
  <event eventID="2988">
    <action>Create</action>
    <userID>adminUser</userID>
    <notes>Folder Drawings was created by adminUser
    <dateStamp>2006-06-12T14:34:20</dateStamp>
  </event>
</auditTrail>
</folder>
</folders>
</project>
</Enterprise>
```

---

## 6.1.8 Document Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new document object (documentL2N1), within the folder object (folderL1N1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <document documentID="documentL2N1">
          <name>Sub Surface Drainage Overview</name>
          <description>Draft Drainage Layouts</description>
          <status>Active</status>
          <ownerID>AdminUser</ownerID>
        </document>
      </folder>
    </folders>
  </project>
</Enterprise>
```

```

<createdByID>AdminUser</createdByID>
<dateCreated>2006-06-12T15:15:20</dateCreated>
<lastModifiedByID>AdminUser</lastModifiedByID>
<dateLastModified>2006-06-12T15:15:20</dateLastModified>
<acl>
  <permissions originatorID="AdminUser">
    <userID>AdminUser</userID>
    <see>true</see>
    <seeContents>true</seeContents>
    <modify>true</modify>
    <editACL>true</editACL>
    <editAttributes>true</editAttributes>
    <addItem>true</addItem>
    <deleteRevision>true</deleteRevision>
    <delete>true</delete>
    <addRevision>true</addRevision>
  </permissions>
</acl>
<auditTrail>
  <event eventID="3100">
    <action>Create</action>
    <userID>AdminUser</userID>
    <notes>Document Sub Surface Drainage
Overview was created by AdminUser 15:15:20 12 June 2006</notes>
    <dateStamp>2006-06-
12T15:15:20</dateStamp>
  </event>
</auditTrail>
<revisions>
  <revision revisionID="revisionL2N1R1">
    <revisionRef>version 1</revisionRef>
    <revisionNotes>This is the First
Revision</revisionNotes>
    <authorID>AdminUser</authorID>
    <dateAuthored>2006-06-12</dateAuthored>
    <revisionStatus>For
Information</revisionStatus>
    <status>Active</status>
    <isLatestRevision>true</isLatestRevision>
    <createdByID>AdminUser</createdByID>
    <dateCreated>2006-06-
12T15:15:20</dateCreated>
    <lastModifiedByID>AdminUser</lastModifiedByID>
    <dateLastModified>2006-06-
12T15:15:20</dateLastModified>
  </revision>
</revisions>
<acl>
  <permissions
originatorID="AdminUser">
    <userID>AdminUser</userID>
    <see>true</see>
    <seeContents>true</seeContents>
    <modify>true</modify>
    <editACL>true</editACL>
    <editAttributes>true</editAttributes>
    <addItem>true</addItem>
  </permissions>

```

```

<deleteRevision>true</deleteRevision>
<delete>true</delete>
<addRevision>true</addRevision>
</permissions>
</acl>
<files>
<file fileID="fileL2N1R1F1">
<fileName>drainage.dwg</fileName>
<MIMEType></MIMEType>
<fileSize>3265372</fileSize>
<status>Active</status>
<pathToFile>FileStore\fileL2N1R1F1\drainage.dwg</pathToFile>
<auditTrail>
<event
eventID="3102">
<action>Create</action>
<userID>AdminUser</userID>
<notes>File drainage.dwg was created by AdminUser 15:15:20 12 June 2006</notes>
<dateStamp>2006-06-12T15:15:20</dateStamp>
</event>
</auditTrail>
</file>
</files>
<auditTrail>
<event eventID="3101">
<action>Create</action>
<userID>AdminUser</userID>
<notes>Revision version 1
was created by AdminUser 15:15:20 12 June 2006</notes>
<dateStamp>2006-06-
12T15:15:20</dateStamp>
</event>
</auditTrail>
</revision>
</revisions>
</document>
</folder>
</folders>
</project>
</Enterprise>

```

### 6.1.9 Alias Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new alias object (aliasL2N1), within the folder object (folderL1N1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctpl\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <alias aliasID="alalsL2N1">
          <name>Short Cut to Document</name>
          <documentID>documentL2N1</documentID>
        </alias>
      </folder>
    </folders>
  </project>
</Enterprise>
```

---

</folders>  
</project>  
</Enterprise>

### 6.1.10 Revision Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new revision object (revisionL2N1R2), within the document object (documentL2N1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <document documentID="documentL2N1">
          <name>Sub Surface Drainage Overview</name>
          <status>Active</status>
          <createdByID>AdminUser</createdByID>
          <dateCreated>2006-06-12T15:15:20</dateCreated>
        </document>
      </folder>
    </folders>
  </project>
</Enterprise>
```

```

<lastModifiedByID>AdminUser</lastModifiedByID>
<dateLastModified>2006-06-13T21:31:00</dateLastModified>
<acl>
<auditTrail>
  <event eventID="3110">
    <action>Revise</action>
    <userID>AdminUser</userID>
    <notes>Revision 2 was created by AdminUser

21:31:00 13 June 2006</notes>
13T21:31:00</dateStamp>
    <dateStamp>2006-06-

  </event>
</auditTrail>
<revisions>
  <revision revisionID="revisionL2N1R2">
    <revisionRef>version 2</revisionRef>
    <revisionNotes>This is the First
Revision</revisionNotes>
    <authorID>AdminUser</authorID>
    <dateAuthored>2006-06-13</dateAuthored>
    <revisionStatus>For
Information</revisionStatus>
    <status>Active</status>
    <isLatestRevision>true</isLatestRevision>
    <createdByID>AdminUser</createdByID>
    <dateCreated>2006-06-
13T21:31:00</dateCreated>
    <lastModifiedByID>AdminUser</lastModifiedByID>
    <dateLastModified>2006-06-
13T21:31:00</dateLastModified>
    <acl>
      <permissions>
        <originatorID="AdminUser">
          <userID>AdminUser</userID>
          <see>true</see>
          <seeContents>true</seeContents>
          <modify>true</modify>
          <editACL>true</editACL>
          <editAttributes>true</editAttributes>
          <addItem>true</addItem>
          <deleteRevision>true</deleteRevision>
          <delete>true</delete>
          <addRevision>true</addRevision>
        </permissions>
      </acl>
    <files>
      <file fileID="fileL2N1R2F1">
        <fileName>drainagev2.dwg</fileName>
        <MIMEType></MIMEType>
        <fileSize>3721331</fileSize>
        <status>Active</status>

```

# Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

---

```
<pathToFile>FileStore\fileL2N1R2F1\drainagev2.dwg</pathToFile>
<auditTrail>
  <event
    eventID="3112">
      <action>Create</action>
      <userID>AdminUser</userID>
      <notes>File drainagev2.dwg was created by AdminUser 21:31:00 13 June 2006</notes>
      <dateStamp>2006-06-13T21:31:00</dateStamp>
    </event>
  </auditTrail>
</file>
</files>
<auditTrail>
  <event eventID="3111">
    <action>Create</action>
    <userID>AdminUser</userID>
    <notes>Revision version 2
was created by AdminUser 21:31:00 13 June 2006</notes>
    <dateStamp>2006-06-
13T21:31:00</dateStamp>
  </event>
</auditTrail>
</revision>
</revisions>
</document>
  </folder>
</folders>
</project>
</Enterprise>
```

---

### 6.1.11 File Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new file object (fileL2N1R2F2), within the revision object (revisionL2N1R2).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <document documentID="documentL2N1">
          <name>Sub Surface Drainage Overview</name>
          <status>Active</status>
          <createdByID>AdminUser</createdByID>
        </document>
      </folder>
    </folders>
  </project>
</Enterprise>
```

# Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

---

```
<dateCreated>2006-06-12T15:15:20</dateCreated>
<revisions>
  <revision revisionID="revisionL2N1R2">
    <status>Active</status>
    <createdByID>AdminUser</createdByID>
    <dateCreated>2006-06-
13T21:31:00</dateCreated>
    <lastModifiedByID>AdminUser</lastModifiedByID>
    <dateLastModified>2006-06-
13T22:23:00</dateLastModified>
    <files>
      <file fileID="fileL2N1R2F2">
        <fileName>drainagev2.pdf</fileName>
        <MIMEType>application/pdf</MIMEType>
        <fileSize>664456</fileSize>
        <status>Active</status>
        <pathToFile>FileStore\fileL2N1R2F1\drainagev2.pdf</pathToFile>
        <auditTrail>
          <event
eventID="3112">
            <action>Create</action>
            <userID>AdminUser</userID>
            <notes>File drainagev2.pdf was created by AdminUser 22:23:00 13 June 2006</notes>
            <dateStamp>2006-06-13T22:23:00</dateStamp>
          </event>
        </auditTrail>
      </file>
    </files>
    <auditTrail>
      <event eventID="3111">
        <action>Modify</action>
        <userID>AdminUser</userID>
        <notes>File drainagev2.pdf
was added to Revision by AdminUser 22:23:00 13 June 2006</notes>
        <dateStamp>2006-06-
13T21:31:00</dateStamp>
      </event>
    </auditTrail>
  </revision>
</revisions>
</document>
</folder>
</folders>
</project>
</Enterprise>
```

---

## 6.1.12 Recipient Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new recipient object (recipL2N1R2R1), within the revision object (revisionL2N1R2).

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <document documentID="documentL2N1">
          <name>Sub Surface Drainage Overview</name>
          <status>Active</status>
          <createdByID>AdminUser</createdByID>
          <dateCreated>2006-06-12T15:15:20</dateCreated>
        </document>
      </folder>
    </folders>
  </project>
</Enterprise>
```

# Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

---

```
<revisions>
  <revision revisionID="revisionL2N1R2">
    <status>Active</status>
    <createdByID>AdminUser</createdByID>
    <dateCreated>2006-06-
13T21:31:00</dateCreated>
    <lastModifiedByID>AdminUser</lastModifiedByID>
    <dateLastModified>2006-06-
13T22:40:00</dateLastModified>
    <files/>
    <recipients>
      <recipient
recipientID="recipl2N1R2R1">
        <userID>TestUser1</userID>
        <issuedBy>AdminUser</issuedBy>
        <dateIssued>2006-06-
13T22:40:00</dateIssued>
        <status>Active</status>
      </recipient>
    </recipients>
    <auditTrail>
      <event eventID="3121">
        <action>Modify</action>
        <userID>AdminUser</userID>
        <notes>Revision 2 was
Issued To TestUser1 by AdminUser 22:40:00 13 June 2006</notes>
        <dateStamp>2006-06-
13T22:40:00</dateStamp>
      </event>
    </auditTrail>
  </revision>
</revisions>
</document>
</folder>
</folders>
</project>
</Enterprise>
```

---

### 6.1.13 RevisionLink Creation Example XML

---

a) The following example xml file shows the incremental update required for the creation of a new revisionLink object (RL-0000001), within the revision object (revisionL2N1R2).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <document documentID="documentL2N1">
          <name>Sub Surface Drainage Overview</name>
          <status>Active</status>
          <createdByID>AdminUser</createdByID>
          <dateCreated>2006-06-12T15:15:20</dateCreated>
        </document>
      </folder>
    </folders>
  </project>
</Enterprise>
```

Improving Interoperability of AEC Collaborative software through the creation of Data Exchange Standards

---

```

    <revisions>
      <revision revisionID="revisionL2N1R2">
        <status>Active</status>
        <createdByID>AdminUser</createdByID>
        <dateCreated>2006-06-
13T21:31:00</dateCreated>
        <lastModifiedByID>AdminUser</lastModifiedByID>
        <dateLastModified>2006-06-
13T23:03:00</dateLastModified>
        <files/>
        <revisionLinks>
          <revisionLink revisionLinkID="RL-
0000001">
            <childRevisionID>revisionL2N1R1</childRevisionID>
            <status>Active</status>
          </revisionLink>
        </revisionLinks>
        <auditTrail>
          <event eventID="3121">
            <action>Modify</action>
            <userID>AdminUser</userID>
            <notes>RevisionLink to
revisionL2N1R1 was created by AdminUser 23:03:00 13 June 2006</notes>
            <dateStamp>2006-06-
13T23:03:00</dateStamp>
          </event>
        </auditTrail>
      </revision>
    </revisions>
  </document>
</folder>
</folders>
</project>
</Enterprise>

```

## 6.1.14 RevisionStatus Creation Example XML

a) The following example xml file shows the incremental update required for the creation of a new revisionStsus object (revStatus6), within the project object (proj12345).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <revisionStatuses>
      <revisionStatus statusID="revStatus6">
        <description>For Information</description>
        <status>Active</status>
      </revisionStatus>
    </revisionStatuses>
    <acl/>
    <folders/>
  </project>
</Enterprise>
```

## **6.2 Object Modification**

---

The following example xml documents illustrate how various ncctp objects are modified within an incremental update. These files are for illustrate purposes only as likely incremental updates will contain a number of creation, modification, deletion and translations actions.

---

## 6.2.1 Project Modification Example XML

---

a) The following example xml file illustrates the updating of the projectNotes property for a project (project12345).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <projectNotes>Updated Project Notes: these project notes need to be updated on all destination
systems</projectNotes>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders/>
  </project>
</Enterprise>
```

## 6.2.2 Organisation Modification Example XML

---

a) The following example xml file shows the updating of the name property for an organisation object (organ1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations>
      <organisation organisationID="organ1">
        <name>Organisation Alpha Renamed</name>
        <status>Active</status>
        <dateLastModified>2006-04-01T22:19:00</dateLastModified>
        <users/>
        <offices/>
      </organisation>
    </organisations>
    <groups/>
    <ad/>
    <folders/>
  </project>
```

---

<Enterprise>

## 6.2.3 User Modification Example XML

---

a) The following example xml file shows the updating of the phoneNumber property for a user object (user5).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations>
      <organisation organisationID="organ1">
        <name>Organisation Alpha</name>
        <status>Active</status>
        <users>
          <user userID="user5">
            <loginName>testUser5</loginName>
            <firstName>Test</firstName>
            <lastName>User</lastName>
            <email>test.user5@ncctp.net</email>
            <phoneNumber>01509806000</phoneNumber>
            <status>Active</status>
          </user>
        </users>
      </organisation>
    </organisations>
  </project>
</Enterprise>
```

---

```
    </users>
  </offices/>
</organisation>
</organisations>
</groups/>
</acl/>
</folders/>
</project>
</Enterprise>
```

## 6.2.4 Office Modification Example XML

---

a) The following example xml file shows the update of office (office2) property belonging to an organisation (organ1). It shows the updating of the phoneNumber property.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations>
      <organisation organisationID="organ1">
        <name>Organisation Alpha</name>
        <status>Active</status>
        <users/>
        <offices>
          <office officeID="office2">
            <isMainOffice>>false</isMainOffice>
            <phoneNumber>01509111434</phoneNumber>
            <address>
              <address1>address1</address1>
              <town>town</town>
            </address>
          </office>
        </offices>
      </organisation>
    </organisations>
  </project>
</Enterprise>
```

---

```
</address>
</office>
</offices>
</organisation>
</organisations>
</groups/>
</acl/>
</folders/>
</project>
</Enterprise>
```

## 6.2.5 Group Modification Example XML

---

a) The following example xml file illustrates the updating of the name property for a group (group1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectId="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups>
      <group groupId="group1">
        <name>New Group Renamed</name>
        <status>Active</status>
        <groupUsers/>
      </group>
    </groups>
    <acl/>
    <folders/>
  </project>
</Enterprise>
```

---

## 6.2.6 Address Modification Example XML

---

a) The following example xml file shows the update of an address property of an office (office2) of an organisation (organ1). It shows the updating of the zip-postCode property.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations>
      <organisation organisationID="organ1">
        <name>Organisation Alpha</name>
        <status>Active</status>
        <users>
        <offices>
          <office officeID="office2">
            <isMainOffice>>false</isMainOffice>
            <address>
              <address1>address1</address1>
              <town>town</town>
              <zip-postCode>HP100PB</zip-postCode>
            </address>
          </office>
        </offices>
      </organisation>
    </organisations>
  </project>
</Enterprise>
```

```

        </address>
      </office>
    </offices>
  </organisation>
</organisations>
</groups/>
<acl/>
</folders/>
</project>
</Enterprise>

```

b) The following example xml file shows the update of an address object that is the child of a project (proj12345) object. It shows the updating all address object properties.

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <address>

```

---

```
<address1>address1 Modified</address1>
<address2>address2 Modified</address1>
<address3>address3 Modified</address1>
<town>town Modified</town>
<state-County>changed</state-County>
<country>changed</country>
<zip-postCode>HP109PN</zip-postCode>
</address>
<acl/>
<folders/>
</project>
</Enterprise>
```

## 6.2.7 Folder Modification Example XML

---

a) The following example xml file shows the update of the folder object property, name and its associated audit entry.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctpl\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <lastModifiedByID>AdminUser</lastModifiedByID>
        <dateLastModified>2006-06-12T14:51:37</dateLastModified>
        <auditTrail>
          <event eventID="2989">
            <action>Modify</action>
            <userID>adminUser</userID>
          </event>
        </auditTrail>
      </folder>
    </folders>
  </project>
</Enterprise>
```

---

adminUser 14:51:37 12 June 2006</notes>

<notes>Name changed to Project Drawings by

<dateStamp>2006-06-12T14:51:37</dateStamp>

</event>

</auditTrail>

</folder>

</folders>

</project>

</Enterprise>

## 6.2.8 Document Modification Example XML

---

a) The following example xml file shows the update of the document object property, description and its associated audit entry.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <document documentID="documentL2N1">
          <name>Sub Surface Drainage Overview</name>
          <description>Updated Description</description>
          <status>Active</status>
          <createdByID>AdminUser</createdByID>
          <dateCreated>2006-06-12T15:15:20</dateCreated>
        </document>
      </folder>
    </folders>
  </project>
</Enterprise>
```

---

```
<lastModifiedByID>AdminUser</lastModifiedByID>
<dateLastModified>2006-06-12T16:02:00</dateLastModified>
<auditTrail>
  <event eventID="3110">
    <action>Modify</action>
    <userID>AdminUser</userID>
    <notes>Description was modified by
AdminUser 16:02:00 12 June 2006</notes>
  </event>
  </auditTrail>
  <revisions/>
</document>
</folder>
</folders>
</project>
</Enterprise>
<dateStamp>2006-06-12T16:02:00</dateStamp>
```

## 6.2.9 Alias Modification Example XML

---

a) The following example xml file illustrates the updating of the name property for an alias object (aliasL2N1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <alias aliasID="aliasL2N1">
          <name>Short Cut to Document(Renamed)</name>
          <documentID>documentL2N1</documentID>
        </alias>
      </folder>
    </folders>
  </project>
```

---

</Enterprise>

## 6.2.10 Revision Modification Example XML

---

a) The following example xml file illustrates the updating of the revisionStatus property for a revision (revisionL2N1R2).

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <document documentID="documentL2N1">
          <name>Sub Surface Drainage Overview</name>
          <status>Active</status>
          <createdByID>AdminUser</createdByID>
          <dateCreated>2006-06-12T15:15:20</dateCreated>
          <revisions>

```

```

<revision revisionID="revisionL2N1R2">
  <revisionStatus>For
    <status>Active</status>
    <createdByID>AdminUser</createdByID>
    <dateCreated>2006-06-
      13T21:31:00</dateCreated>
      <lastModifiedByID>AdminUser</lastModifiedByID>
      <dateLastModified>2006-06-
        13T21:44:00</dateLastModified>
        <files/>
        <auditTrail>
          <event eventID="3121">
            <action>Create</action>
            <userID>AdminUser</userID>
            <notes>Revision 2
            revisionStatus Changed to For Comment by AdminUser 21:44:00 13 June 2006</notes>
            <dateStamp>2006-06-
              13T21:44:00</dateStamp>
              </event>
            </auditTrail>
          </revision>
        </revisions>
      </document>
    </folder>
  </folders>
</project>
</Enterprise>

```

### 6.2.11 File Modification Example XML

---

a) The following example xml file illustrates the updating of the fileSize property for a file object (fileL2N1R2F2).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <document documentID="documentL2N1">
          <name>Sub Surface Drainage Overview</name>
          <status>Active</status>
          <createdByID>AdminUser</createdByID>
          <dateCreated>2006-06-12T15:15:20</dateCreated>
        </document>
      </folder>
    </folders>
  </project>
</Enterprise>
```

```

    <revisions>
      <revision revisionID="revisionL2N1R2">
        <status>Active</status>
        <createdByID>AdminUser</createdByID>
        <dateCreated>2006-06-
13T21:31:00</dateCreated>
        <files>
          <file fileID="fileL2N1R2F2">
            <fileName>drainagev2.pdf</fileName>
            <fileSize>1000564</fileSize>
            <status>Active</status>
            <pathToFile/>
            <auditTrail>
              <event
eventIID="3112">
                <action>Modify</action>
                <userID>AdminUser</userID>
                <notes>File drainagev2.pdf was Updated by AdminUser 22:31:00 13 June 2006</notes>
                <dateStamp>2006-06-13T22:31:00</dateStamp>
              </event>
            </auditTrail>
          </file>
        </files>
      </revision>
    </revisions>
  </document>
</folder>
</folders>
</project>
</Enterprise>

```

## 6.2.12 Recipient Modification Example XML

---

a) The following example xml file illustrates the updating of the dateAcknowledged property for a recipient (recipL2N1R2R1).

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <acl/>
    <folders>
      <folder folderID="folderL1N1">
        <name>Project Drawings</name>
        <status>Active</status>
        <document documentID="documentL2N1">
          <name>Sub Surface Drainage Overview</name>
          <status>Active</status>
          <createdByID>AdminUser</createdByID>
          <dateCreated>2006-06-12T15:15:20</dateCreated>
          <revisions>

```

```

<revision revisionID="revisionL2N1R2">
  <status>Active</status>
  <createdByID>AdminUser</createdByID>
  <dateCreated>2006-06-
13T21:31:00</dateCreated>
  <lastModifiedByID>AdminUser</lastModifiedByID>
  <dateLastModified>2006-06-
13T22:40:00</dateLastModified>
  <files/>
  <recipients>
    <recipient
recipientID="recipl2N1R2R1">
  <userID>TestUser1</userID>
  <issuedBy>AdminUser</issuedBy>
  <dateAcknowledged>2006-
06-13T22:44:00</dateAcknowledged>
  <status>Active</status>
  </recipient>
</recipients>
  <auditTrail>
    <event eventID="3122">
      <action>Modify</action>
      <notes>Revision 2 was
Acknowledged by TestUser1 by 22:44:00 13 June 2006</notes>
      <dateStamp>2006-06-
13T22:44:00</dateStamp>
    </event>
  </auditTrail>
</revision>
</revisions>
</document>
  </folder>
</folders>
</project>
</Enterprise>

```

### 6.2.13 RevisionStatus Modification Example XML

---

a) The following example xml file illustrates the updating of the description property for a revisionStatus Object (revStatus6).

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Sample XML file generated by XMLSpy v2005 U (http://www.xmlspy.com)-->
<Enterprise xmlns="http://www.ncctp.net" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ncctp.net
C:\DOCUME~1\SCOTTM~1\Desktop\ncctp\NCCTP_INC_v1_000.xsd">
  <project projectID="proj12345">
    <exportMetadata>
      <featuresExported>
        <project>true</project>
        <organisations>true</organisations>
        <users>true</users>
        <offices>true</offices>
        <projectAddress>true</projectAddress>
        <userAddress>true</userAddress>
        <officeAddress>true</officeAddress>
        <groups>true</groups>
        <revisionStatuses>true</revisionStatuses>
        <projectACL>true</projectACL>
        <folderACL>true</folderACL>
        <documentACL>true</documentACL>
        <revisionACL>true</revisionACL>
        <documentAuditTrail>true</documentAuditTrail>
        <revisionAuditTrail>true</revisionAuditTrail>
        <fileAuditTrail>true</fileAuditTrail>
        <revisionLinks>true</revisionLinks>
        <revisionRecipients>true</revisionRecipients>
        <folders>true</folders>
        <documents>true</documents>
        <aliases>true</aliases>
        <revisions>true</revisions>
        <files>true</files>
        <multipleFiles>true</multipleFiles>
      </featuresExported>
      <dateOfExport>2006-04-01</dateOfExport>
      <exportNotes>Incremental Update</exportNotes>
      <sourceSystem>CC</sourceSystem>
    </exportMetadata>
    <name>NCCTP Incremental Update Project</name>
    <status>Active</status>
    <clientID>Technical Group (NCCTP)</clientID>
    <organisations/>
    <groups/>
    <revisionStatuses>
      <revisionStatus statusID="revStatus6">
        <description>Information</description>
        <status>Active</status>
      </revisionStatus>
    </revisionStatuses>
    <acl/>
    <folders/>
  </project>
</Enterprise>
```

---

### **6.3 Object Deletion**

---

The following example xml documents illustrate how various ncctp objects are deleted within an incremental update. These files are for illustrate purposes only as likely incremental updates will contain a number of creation, modification, deletion and translations actions.

## **6.4 Object Translation**

---

The following example xml documents illustrate how various ncctp objects are translated within an incremental update. These files are for illustrate purposes only as likely incremental updates will contain a number of creation, modification, deletion and translations actions.

---

## **APPENDIX I NCCTP REAL TIME INTEGRATION SPECIFICATION**

---

NCCTP                      Real-Time  
Collaborative              System  
Integration  
Specification [DRAFT]

---

NCCTP Specification Request 3  
Version 1.0  
14 November 2006

## **Acknowledgements**

This specification is the collaborative product of

Scott Moses (specification lead, Causeway),

Scott Moses (author, Causeway)

And other members of the NCCTP expert Group, including

4Projects

Aconex

Asite

BIW

Business Collaborator

CadWeb

Causeway Technologies

Sarcophagus

---

## **License**

This specification may only currently be used by NCCTP member organisations.

---

## 1 Preface

---

This is version 1.0 of the NCCTP Real-Time Collaborative System Integration Specification.

### **1.1 Documents Included**

---

This specification includes:

- This document in Microsoft Word Format (ncctpRTiv1.0.doc).
- The XML Schema Document that defines the data transferred (RTiv1.0.xsd).

In case of discrepancy between this document and the contents of the RTiv1.0 XML Schema this document should be considered normative.

---

---

## 2 Introduction

---

### 2.1 Motivation

---

The UK AEC marketplace for collaborative software has a number of different suppliers, with solutions picked by organisations or on a per project basis. This scenario results in users from a single organisation using a multitude of different systems across numerous projects. The motivation behind this initiative by the NCCTP is to allow a user of multiple collaborative systems, the ability to interact with content stored in them through a single familiar user interface.

Since members of the NCCTP have already constructed a generic way of describing the contents which exist within an AEC Specific collaborative system, a generic set of methods to access and update this information is a logical extension.

Using a real-time based integration solution between vendors, there will be no requirement to duplicate repository content, allowing users accesses to the current information when it is added to any of the collaborating systems.

### 2.2 Goals

---

The guiding principles governing the design of this specification are:

**It should not be tied to any particular underlying architecture, data source or protocol.**

The specification is an extension of the NCCTP bulk data exchange specification, which describes the commonalities which exist between different project collaboration systems used by the AEC. The primary challenge is to ensure that any proposed solution is flexible enough to be applied to none NCCTP members should they join the organisation.

**It should allow for relatively easy implementation on top of as wide a variety of existing systems as possible.**

The solution for real-time integration must cover only the core functionality of typical collaboration systems used by the AEC sector. This is to enable the effort put into deploying the solution, generates the must reward for end users of these collaborative tools.

**It should transfer the minimum data between systems directly.**

To minimise the amount of traffic between the collaborative systems only data must be included, not presentational information. The presentation layer placed on top of the data will be dependant upon the collaborative service the user is accessing.

---

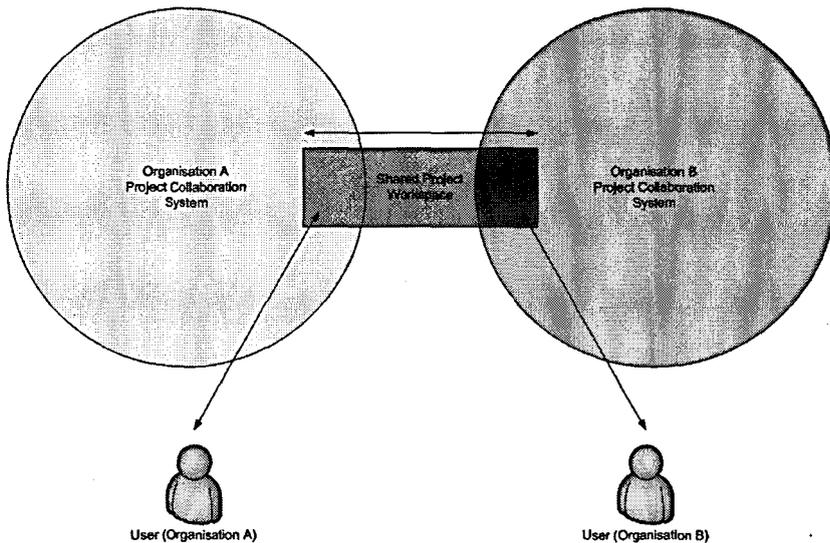
## 3 Use Cases

---

### 3.1 Large Project Joint Venture

---

Main Contractors, UK Construction and Highway Build have undertaken a joint venture to improve and maintain a major road in the UK. Both organisations have primary used a single collaborative solution for their projects over the past few years and users are familiar with how the applications work after use, and training. In the current environment the two contractors would need to decide upon which collaborative product would be used for the joint venture and the staff of one or both if a 3<sup>rd</sup> collaborative solution was selected would need retraining and time to become familiar.



The two contractors decide to implement the NCCTP real-time collaborative system integration solution, allowing multiple repositories to be linked for a single project. This linkage between the systems allows users of each the ability to access content added to other systems through a familiar user interface as if the content had been added directly to their own system.

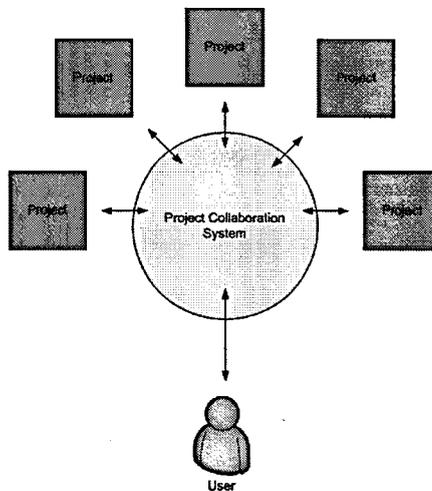
Through using a real-time integration solution both organisation are able to conduct the collaborative work in

a environment where they have access to the wealth of information which already exists in their own systems, and expose the collaborative content to other users without the need to specifically migrate the data into the system.

### **3.1 AEC Organisation collaborating on multiple projects**

A sub contractor works on a large number of concurrent projects for many different clients and which includes having to communicate project information through a large number of different collaborative tools. Therefore employees of the company need to know the basics of lots of different collaborative systems, and the company's employees have no mechanism to interrogate data across projects. To solve this and other problems the company is investing in an enterprise collaboration solution to store project data.

To enable project information which will continue to be stored on numerous external collaborative systems to be accessed via this new enterprise wide collaborative system the company have decided to utilise the NCCTP real-time integration specification.



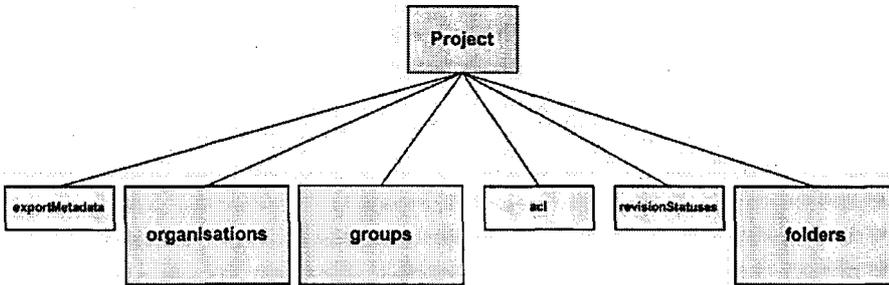
Through its implementation the specification will allow the organisation users to interact with project information stored in a number of different repositories without the need to leave the enterprise wide collaborative system.



## 4.1 Major Repository Branches

---

The Content Repository is split into three major branches; the first holds the information relating to project users, the second holds information relating to user collections within the project, while the third stores the project documents.

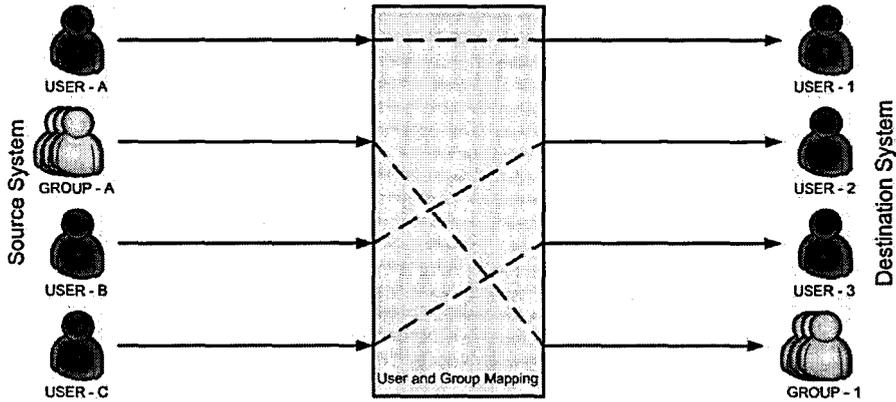


### 4.1.1 Project Users and Project Groups

---

For project data to be successfully accessed from multiple collaborative systems the same user accounts must exist on all integrated systems, to ensure that access control lists are maintained. Data transferred in real-time from other collaborative systems will contain references to user and group objects that exist within the source systems content repository. These references will need to be translated by the destination system to users and groups objects that exist within its content repository.

To allow these translations the destination system is required to keep user and group object mapping data; While the source system is also required to maintain that mapping data so the correct information is passed through to the collaborative system generating the request.



#### 4.1.2 Project Documents

Project documents refer to documents and the filing structure in which they are stored in the content repository.

Since the primary use of real-time integration of collaborative systems is to allow.

To allow for translations to be made the destination system is required to keep object mapping data; while the source system is required to maintain that data through the incremental updates it passes to the destination system.

---

## 5 Real Time Integration

---

Real time integration or real time access is the ability to interact with content which is stored in different collaborative system then the user is accessing by. Differing from proposed NCCTP incremental integration it can allow users from both systems to manipulate the project data since documents, revisions, and files are only stored in a single system.

In a real time integration scenario all audit history and access control list changes are stored in the system were the object resides. The audit information stored in the system can be viewed, but not updated from another system. Since all interaction with objects in the system will be conducted through the integration then it is the system were the object resides which must handle all audit recording.

Real time integration between systems will appear to the end users as seamless access to all information regardless of where it resides without the need to authenticate to different systems.

It is presented as a series of API's which define the functionality which the source system must provide to all destination systems.

### 5.1 Authentication

---

To enable permissions to be applied across multiple different collaborative systems the same user and group structure must exist on all systems. For example the users and groups which are participating on a project in one system must also be defined and participating on the integrated project on the other systems.

Users accessing a 3<sup>rd</sup> collaborative system through their own collaborative system will not communicate directly with the other system. All their requests will be sent to the same system and then forwarded by that system to the collaborative system were the data resides. Therefore it will not be required to enable user/application authentication only application/application authentication.

---

To enable the receiving application to know which user is requesting the data, each call made to it must include encrypted authentication information. The user authentication information can then be read and the correct information returned based upon that user's permissions.

## **5.2 User and Group Synchronisation**

---

Between all participating collaborative systems, users and groups associated with the shared project must be the same in all systems. Since a degree of manipulation of the project participants can occur on any of the linked systems it is recommended that separate groups are created for this project which do not interact with other projects stored on the collaborative instance.

The synchronisation of user and group information between the collaborative systems should be based on the transactions outlined in the NCCTP Incremental Project Specification.

The Adjustment of User details must only be conducted on the system where the user was initially created, i.e. the system which the user accesses the collaborative cluster through. This is to avoid potential conflicts which could occur if details were adjusted in multiple systems at the same time.

## **5.3 Repository Browsing**

---

Repository browsing functions are based around the user's ability to navigate through the structure of a collaborative project. These functions will allow for the user to browse inside the project, its folders, examine the revisions of a document and the files associated with any revision. The information displayed to the user will always be dependant upon the permissions that user has on the objects they are examining.

### **5.3.1 Project Object Browsing**

---

This function will return a list of objects which are direct children of the project object, based upon the rights of the requesting user. To allow the correct information to be returned to the user both user authentication information and the id of the project object must be specified in the request.

The response return will include a list of objects contained within the project object, each of which will include the metadata defined by the NCCTP Bulk Data Exchange Standard. The response should not contain any nested sub information.

The results returned from the call, will contain data only and should be rendered by the requesting application and presented to the user. Should no results be returned or and error occur the requesting application is responsible for informing the user of what has happened.

Since every collaborative system in the cluster will have its own project object it is not expected that this call between systems will actual be used in practice, and is included for completeness.

### **5.3.2 Folder Object Browsing**

---

This function will return a list of objects, which are direct children of the folder object, based upon the rights of the requesting user. To allow a response to be generated by the system both user authentication and the id of the folder must be supplied in the request. Using the user information supplied in the request the application should only return those objects which the specified user has rights to see.

The response returned will include a list of zero to many objects contained within the folder object, each of which will include the metadata defined by the NCCTP Bulk Data Exchange Standard. The objects included in response should not contain any nested sub information.

The results returned from the call, will contain data only and should be rendered by the requesting application and presented to the user. Should no results be returned or

---

should an error occur the requesting application is responsible for informing the user of what has occurred.

### **5.3.3 Document Object Browsing**

---

This function will return a list of document revisions, which belong to the document object. To allow a response to be generated by the system both user authentication and the id of the document must be supplied in the request. Using the user information supplied in the request the application should only return those objects which the specified user has rights to see. Filtering of document revisions will only be done if the application supports permissions stored at the revision level.

The response returned will include a list of one to many document revisions contained within the document object, each of which will include the metadata defined by the NCCTP Bulk Data Exchange Standard. The objects included in response should not contain any nested sub information.

The results returned from the call, will contain data only and should be rendered by the requesting application and presented to the user. Should no results be returned or if an error is generated by the request the requesting application is responsible for informing the user of what has occurred.

### **5.3.3 Document Revision Object Browsing**

---

This function will return a list of files which belong to the document revision object. To allow a response to be generated by the system both user authentication and the id of the document revision must be supplied in the request. Using the user information supplied in the request the application should only return those objects which the specified user has rights to see. Filtering of files will only be done if the application supports permissions stored at the file level; else all files will be return.

The response returned will include a list of one to many files contained in the document revision object, each of which will include the metadata defined by the NCCTP Bulk Data Exchange Standard. The objects included in response should not contain any nested sub information.

The results returned from the call, will contain data only and should be rendered by the requesting application and presented to the user. Should no results be returned or an error be generated the requesting application is responsible for informing the user.

## **5.4 View Object Metadata**

---

View Object Metadata functions allow information about specified objects to be retrieved from any of the collaborative systems in the cluster. This function can be executed against any object contained in the collaborative system defined by the NCCTP in the Bulk Data Exchange Specification.

### **5.4.1 View Project Object Metadata**

---

This function will return the metadata associated with the project object requested. To allow for a response containing metadata to be returned the request must contain both user authentication information and the id of the project object for which the metadata is required. Return of the project object's metadata will be dependant upon the user's permission to view the metadata of the project object.

The response returned by the function will include the metadata defined by the NCCTP Bulk Data Exchange Standard for the project object. The response will not contain any of the nested information about other sub objects which are contained in the project.

The results returned from the call, will contain data only and should be rendered by the requesting application and presented to the user. Should no results be returned or if an error occurs the requesting application is responsible for informing the user of what has occurred.

Since the project object will reside in every system it is not expected that this call will be used in real life, it is however included for completeness. View project object metadata should be handled by the system which the user is accessing directly.

### **5.4.2 View Folder Object Metadata**

---

---

This function will return the metadata associated with the folder object requested. To allow for a response containing metadata to be returned the request must contain both user authentication information and the id of the folder object for which the metadata is required. Return of the folder object's metadata will be dependant upon the user's permission to view the metadata of the folder object.

The response returned by the function will include the metadata defined by the NCCTP Bulk Data Exchange Standard for the folder object. The response will not contain any of the nested information about other sub objects which are contained in the folder.

The results returned from the call, will contain data only and should be rendered by the requesting application and presented to the user. Should no results be returned or an error be generated the requesting application is responsible for informing the user of what has occurred, based upon the information returned by the other collaborative system or lack of response.

### **5.4.3 View Document Object Metadata**

---

This function will return the metadata associated with the document object requested. To allow for a response containing metadata to be returned the request must contain both user authentication information and the id of the document object for which the metadata is required. Return of the document object's metadata will be dependant upon the user's permission to view the metadata of the document object.

The response returned by the function will include the metadata defined by the NCCTP Bulk Data Exchange Standard for the document object. The response will not contain any of the nested information about document revisions which are contained in the document.

The results returned from the call, will contain data only and should be rendered by the requesting application and presented to the user. Should no results be returned or if an error occurs the requesting application is responsible for informing the user of what has occurred, based upon the information returned by the other collaborative system or the lack of response.

#### **5.4.4 View Document Revision Object Metadata**

---

This function will return the metadata associated with the document revision object requested. To allow for a response containing metadata to be returned the request must contain both user authentication information and the id of the document revisions object for which the metadata is required. Return of the document revisions object's metadata will be dependant upon the user's permission to view the metadata of the document revision object, or permissions on the parent document if not held at the document revision level.

The response returned by the function will include the metadata defined by the NCCTP Bulk Data Exchange Standard for the document revision object. The response will not contain any of the nested information about files which are contained in the document revision.

The results returned from the call, will contain data only and should be rendered by the requesting application and presented to the user. Should no results be returned or if an error occurs the requesting application is responsible for informing the user of what has occurred, based upon the information returned by the other collaborative system or the lack of response.

#### **5.4.5 View File Object Metadata**

---

This function will return the metadata associated with the file object requested. To allow for a response containing metadata to be returned the request must contain both user authentication information and the id of the file object for which the metadata is required. Return of the file object's metadata will be dependant upon the user's permission to view the metadata of the file object, or permissions on the parent document/document revision if not held at the file level.

The response returned by the function will include the metadata defined by the NCCTP Bulk Data Exchange Standard for the file object. The response will not contain the physical file which is associated with the file object.

The results returned from the call, will contain data only and should be rendered by the requesting application and

---

presented to the user. Should no results be returned or if an error occurs the requesting application is responsible for informing the user of what has occurred, based upon the information returned by the other collaborative system or the lack of response.

## **5.5 Update Object Metadata**

---

The update object metadata functions allow for objects which do not exist in the system which the user is interacting with directly to have their metadata updated. This type of function can be executed against any of the objects defined in the NCCTP Bulk Data Exchange Standard.

### **5.5.1 Update Project Object Metadata**

---

Allows some of the metadata held against the project object to be updated from any of the collaborative systems which are participating on the project. To allow an update to be successfully executed both user authentication information and id of the project object must be passed in the request along with the updated metadata for the project object. The update of project metadata will be dependant upon the users permissions on the project object which they are attempting to update.

The generated response will include a message of success or the reason why the update failed. It is the responsibility of the collaborative system making the request to inform the user of either its success or failure.

Since the project object will be stored on every one of the collaborative systems contained in the cluster this call should not be used. Updates to the metadata on the project object should be handled locally by the collaborative system the user is interacting with and then replicated across the entire cluster.

### **5.5.2 Update Folder Object Metadata**

---

Allows some of the metadata held against a folder object to be updated. To allow for an update to be successfully executed both user authentication information and id of the folder object must be passed in the request along with the updated metadata for the folder object. The

update of folder metadata will be dependant upon the users permissions on the folder object which they are attempting to update. Also the metadata selected for update must not be one of the restricted items.

The generated response will include a message of success or the reason why the update failed. It is the responsibility of the collaborative system making the request to inform the user of either its success or failure.

### **5.5.3 Update Document Object Metadata**

Allows some of the metadata held against a document object to be updated. To allow for an update to be successfully executed both user authentication information and id of the document object must be passed in the request along with the updated metadata for the document object. The update of document metadata will be dependant upon the user's permissions on the document object which they are attempting to update, and the metadata selected for update which must not be one of the restricted metadata items.

The generated response will include a message of success or the reason why the update failed. It is the responsibility of the collaborative system making the request to inform the user of either its success or failure.

---

#### **5.5.4 Update Document Revision Object Metadata**

---

Allows some of the metadata held against a document revision object to be updated. To allow for an update to be successfully executed both user authentication information and id of the document revision object must be passed in the request along with the updated metadata for the document revision object. The update of document revision metadata will be dependant upon the user's permissions on the document revision object or on the parent document if not held at the document revision level. Also the metadata selected for update which must not be one of the restricted metadata items.

The generated response will include a message of success or the reason why the update failed. It is the responsibility of the collaborative system making the request to inform the user of either its success or failure.

#### **5.5.5 Update File Object Metadata**

---

Allows some of the metadata held against a file object to be updated. To facilitate a successfully update of file metadata to be executed both user authentication information and id of the file object must be passed in the request along with the updated metadata for the file object. The update of file metadata will be dependant upon the user's permissions on the file object or on the parent document revision / document if not held at the file level. Also the metadata selected for update which must not be one of the restricted metadata items on the file object.

The generated response will include a message of success or the reason why the update failed. It is the responsibility of the collaborative system making the request to inform the user of either its success or failure.

## **5.6 Object Creation**

---

The set of object creation methods contained in the Real time specification allow for objects to be created in container objects which do not reside on the same system as the user is interacting directly with. For example a user wants to add a document to a folder, but the folder does not reside on their system.

Object Creation can be used on any of the objects specified in the NCCTP Bulk Data Exchange Standard, and are outlined in the following sections.

### **5.6.1 Project Creation**

---

This function allows for a project object to be created in collaborative system by a user who is not accessing that system directly. To enable the project creation request to be successfully executed, the full incremental project object definition must be passed in the request, along with user authentication information and the id of the object to which the project is going to be added as a child. The creation of the project object will depend upon the user's rights to add objects to the specified container object, and that the definition of the project object contained in the request is complete.

The generated response will include a message of success, along with the generated id for the project or the reason why the create request has failed. It is the responsibility of the collaborative system making the request to inform the user of either its success or failure.

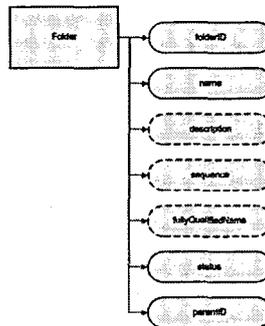
Since all collaborative systems in the cluster are working on a single project which must have been created prior to the commencement of real time updates, this request should never be made. It has been included for completeness and to support the future concept of sub projects inside the main project.

---

### 5.6.2 Folder Creation

---

This function allows for a folder object to be created in collaborative system by a user who is not directly accessing that system. To enable the folder creation request to be successfully executed, the full incremental folder object definition must be passed in the request, along with user authentication information and the id of the object to which the folder is going to be added as a child. The folder definition passed in this request should not contain any additional child objects since the request should be sent in real time as the user attempts to create the folder. The creation of the folder object will depend upon the user's rights to add objects to the specified container object, and that the definition of the folder object contained in the request is valid.



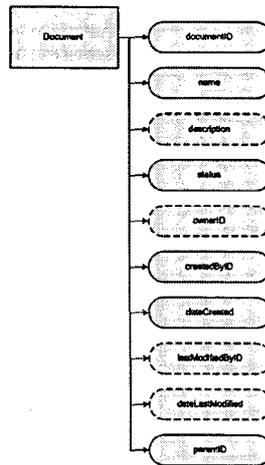
The generated response will include a message of success, along with the generated id for the folder or the reason why the create request has failed. It is the responsibility of the collaborative system making the request to inform the user of either its success or failure.

### 5.6.3 Document Creation

---

This function allows for a document object to be created in collaborative system by a user who is accessing via another collaborative system. To enable the document creation request to be successfully executed, the full incremental document object definition must be passed in the request, along with user authentication information and the id of the object to which the document is going to be added as a child. The document definition passed in this request should not contain any additional child revisions which should be sent in additional requests from

the application. The creation of the document object will depend upon the user's rights to add objects to the specified container object, and that the definition of the document object contained in the request is valid.

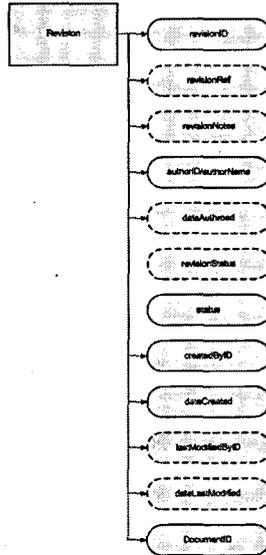


The generated response will include a message of success, along with the generated id for the document or the reason why the create request has failed. It is the responsibility of the collaborative system making the request to inform the user of either its success or failure.

#### **5.6.4 Document Revision Creation**

---

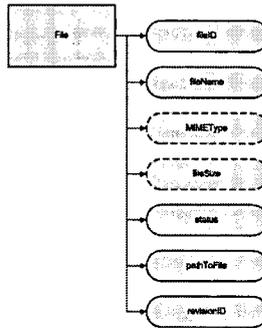
This function allows for a document revision object to be added to a document in collaborative system by a user who is not directly accessing the system. To enable the document revision creation request to be successfully executed, the full incremental document revision object definition must be passed in the request, along with user authentication information and the id of the document to which the document revision is going to be added. The document revision definition passed in this request should not contain any files which should be sent in additional requests from the application. The creation of the document revision object will depend upon the user's rights to add objects to the specified document object, and that the definition of the document revision object contained in the request is correct.



The generated response will include a message of success, along with the generated id for the document revision or the reason why the create request has failed. It is the responsibility of the collaborative system making the request to inform the user of either its success or failure.

### 5.6.5 File Creation

This function allows for a file object to be added to a document revision in collaborative system by a user who is not directly accessing the system. To enable the file creation request to be successfully executed, the full incremental file object definition must be passed in the request, along with user authentication information and the id of the document revision to which the file is going to be added. The file definition passed in this request should not contain the physical file, which will only be referenced inside the request, standard NCCTP behaviour. The creation of the file object will depend upon the user's rights to add files to the specified document revision object, and that the definition of the file object contained in the request is valid.



The generated response will include a message of success, along with the generated id for the file or the reason why the create request has failed. It is the responsibility of the collaborative system making the request to inform the user of either its success or failure.

## 5.7 Object Deletion

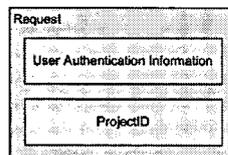
---

The set of object deletion methods allow for objects to be removed from the collaborative environment, although they do not reside in the same repository as the user is accessing. Object deletion can be used on any of the objects specified in the NCCTP Bulk Data Exchange Standard, and are outlined in the following sections.

### 5.7.1 Project Object Deletion

---

This call instructs the collaborative system to delete the specified project object. To allow for the request to be executed both the id of the project object and the user authentication information must be passed in the request. To perform a successful deletion the requesting user must have the rights to delete the project object. Deleting the project object will cause all sub objects contained in the project to be deleted so the user will also require rights to delete all sub objects contained in the project object.



The response generated by the requested system will indicate the success of the request or the reason why the

---

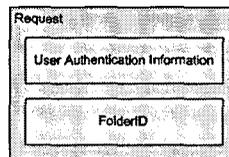
request has failed to execute correctly. It is the responsibility of the requesting system to present the outcome of the request to the user who executed it.

Since the project object is essential to the real time integration the method should not be requested or implemented.

### **5.7.2 Folder Object Deletion**

---

This call instructs the collaborative system to delete the specified folder object. To allow for the request to be executed both the id of the folder object and the user authentication information must be passed in the request. To perform a successful deletion the requesting user must have the rights to delete the folder object. Deleting the folder object will cause all sub objects contained in the folder to be deleted so the user will also require rights to delete all sub objects contained in the folder object.

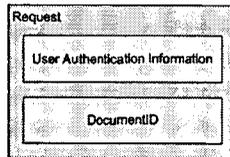


The response generated by the requested system will indicate the success of the request or the reason why the request has failed to execute correctly. It is the responsibility of the requesting system to present the outcome of the request to the user who executed it.

### **5.7.3 Document Object Deletion**

---

This call instructs the collaborative system to delete the specified document object. To allow for the request to be executed both the id of the document object and the user authentication information must be passed in the request. To perform a successful deletion the requesting user must have the rights to delete the document object. Deleting the document object will cause all revisions and files contained in the document to be deleted so the user will also require rights to delete the revisions and files if the system supports permissions at that level.

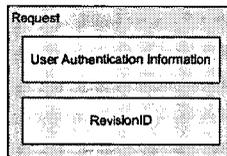


The response generated by the requested system will indicate the success of the request or the reason why the request has failed to execute correctly. It is the responsibility of the requesting system to present the outcome of the request to the user who executed it.

### **5.7.4 Document Revision Object Deletion**

---

This call instructs the collaborative system to delete the specified document revision object. To allow for the request to be executed both the id of the document revision object and the user authentication information must be passed in the request. To perform a successful deletion the requesting user must have the rights to delete the document revision object, or rights to delete from the document if not held at this level. Deleting the document revision object will cause all files contained in the document revision to be deleted so the user will also require rights to delete the files.



The response generated by the requested system will indicate the success of the request or the reason why the request has failed to execute correctly. It is the responsibility of the requesting system to present the outcome of the request to the user who executed it.

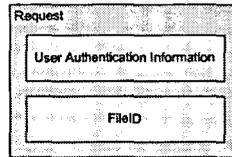
### **5.7.5 File Object Deletion**

---

This call instructs the collaborative system to remove the specified file from a document revision. To allow for the request to be executed both the id of the file object and the user authentication information must be passed in the request. To perform a successful deletion the requesting user must have the rights to delete the file object, or

---

rights to delete from the document or document revision if not held at this level.



The response generated by the requested system will indicate the success of the request or the reason why the request has failed to execute correctly. It is the responsibility of the requesting system to present the outcome of the request to the user who executed it.

### **5.8 Retrieve Physical File**

---

This request retrieves the specified file from a collaborative system which the requesting user is not directly accessing. The request must contain both the id of the file to be retrieved and the user authentication information. To perform a successful fetch the user must have permission to see the file, at the required level for the system hosting the file.

The response generated by the collaborative system will either contain the location of the file which can then be retrieved by the collaborative system the user is accessing or the reason the request has failed. It is the responsibility of the collaborative system that the user is accessing directly to send the file to the user.

