Loughborough University Institutional Repository

Construction project information management in a semantic web environment

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Additional Information:

- A Doctoral Thesis. Submitted in partial fulfillment of the requirements for the award of Doctor of Philosophy of Loughborough University.

Metadata Record: https://dspace.lboro.ac.uk/2134/7823

Publisher: © Jiayi Pan

Please cite the published version.

Loughborough University

This item is held in Loughborough University's Institutional Repository (https://dspace.lboro.ac.uk/) and was harvested from the British Library's EThOS service (http://www.ethos.bl.uk/). It is made available under the following Creative Commons Licence conditions.

creative commons

COMMONS DEED

Attribution-NonCommercial-NoDerivs 2.5

You are free:

- to copy, distribute, display, and perform the work

Under the following conditions:

Attribution. You must attribute the work in the manner specified by the author or licensor.

Noncommercial. You may not use this work for commercial purposes.

No Derivative Works. You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the Legal Code (the full license).

Disclaimer

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

CONSTRUCTION PROJECT INFORMATION MANAGEMENT IN A SEMANTIC WEB ENVIRONMENT

JIAYI PAN

A thesis submitted in partial fulfilment of the requirements
of Loughborough University
for degree of Doctor of Philosophy

November 2006

ABSTRACT

Modern construction projects, characterised by severe fragmentation from both geographical and disciplinary perspectives, require accurate and timely sharing of information. Traditional information management systems operate on a textual basis and do not always consider the meaning of information. Current Web-based information management technology supports information communication to a reasonable extent but still has many limitations, such as the lack of semantic-awareness and poor interoperability of software applications. This research argues that Semantic Web technologies can enhance the efficiency of information management in construction projects by providing content-based and context-specific information to project team members, and supporting the interoperation between independent applications. A Semantic Web-based Information Management System (SWIMS) for construction projects was created to demonstrate the above concept.

The approach adopted for this research involved creating a new framework for Semantic Web-based information management. This extensible system framework enables the system to merge diverse construction information sources, ontologies and end-user applications into the overall Semantic Web environment. The semantic components developed in this research included a project document's annotation model, a project partner's user profile model, and several lightweight IFC-based ontologies for documented information management. This supports intelligent information management and interoperation between heterogeneous information sources and applications. The system framework, prototype annotations, and ontologies were applied to a concept demonstrator that illustrated how the project documents were annotated, accessed, converted, categorised, and retrieved on the basis of content and context. The demonstrator (named SWIMS) acts as a middleware, which mediates between user needs and the information sources. Information in project partners' documents were mapped and accessed intelligently. This involved the use of rule-based filtering and thus prevented the users from being overwhelmed by irrelevant documents or missing relevant ones in heterogeneous and distributed information sources. It also enabled the adaptation of documents to individual contexts and preferences, and the dynamic composition of various document management services.

Evaluation of the system framework and demonstrator revealed that the system enhances the efficiency of construction information management, with the three most beneficial areas being project knowledge management, collaborative design and communication between project team members. The SWIMS annotations, ontologies and deductive rules are important technologies provide an innovative approach to managing construction information. These enable the information in construction documents, both structured documents and un-structured documents, to be interpretable by computers. This ensures the efficiency and precision of construction information management.

Keywords: Semantic Web, Construction Management, Information Management, Ontology, OWL, Document Management

ACKNOWLEDGEMENTS

I would like to express sincere appreciation to my supervisor, Professor Chimay J. Anumba, for his guidance and constant encouragement. He helped me formulate the research objectives and scenarios. Also, he provided inspiration, patient guidance and extensive assistance throughout all research phases. Without his help, this research and the thesis would not be possible.

I also would like to acknowledge:

- Dr. Zhaomin Ren and Dr. Zeeshan Aziz, for sharing their knowledge, which is really useful to this research;
- Mr. Sean T. McAndrew of Taylor Woodrow Construction Ltd. for his support on the use of Extranet and providing practical project documents. Without his help, it would be very hard to accomplish the research;
- Dr. Alistair Duke of BT for his support on the use of Ontoshare;
- The 11 participants who contributed to the research during the evaluation phase for their time and valuable suggestions;
- Protégé-OWL discussion group members for their support on the use of RacerPro and FaCT++.

Finally, and most importantly, I am full of gratitude to my dear Mom and Dad. Their love, support, encouragement and dedication have always been a source of inspiration through all these years. Any achievements would not be possible without them.

GLOSSARY

AEC: Architecture, Engineering, Construction.

Agent: A computer application that makes decisions on behalf of a user but runs without direct human control or constant supervision to accomplish goals provided by a user. An agent typically collects, filters and processes information found on the Web, sometimes with the help of other agents. It is often referred to as an intelligent agent.

Concepts/Classes: sets, collections, or types of objects.

DAML: DARPA Agent Markup Language. It is a preceding Semantic Web language before OWL. DAML program (2006) was sponsored by DARPA's Information Exploitation Office (Defense Advanced Research Projects Agency, USA). It ran from 2000 to 2006 for the development of a Semantic Web language and tools. The Ontology Library of DAML project has collected 282 ontologies covering various domains.

Extranet: A form of network that combines the restricted networks of individual companies or organisations into a wider network-allowing them to transfer documents/messages upon a project.

HTML: HyperText Markup Language. It is a predominant markup language for the creation of Webpages. It provides a means to describe the structure of text-based information in a document — by denoting certain text as headings, paragraphs, lists, and so on — and to supplement that text with interactive forms, embedded images, and other objects.

IAI: International Alliance for Interoperability. It is a not-for-profit organisation that supports the development of the IFC (Industry Foundation Classes) specification.

IFC: Industry Foundation Classes, which is an object-oriented file format with a data model developed by the International Alliance for Interoperability (IAI) to facilitate interoperability in the building industry.

Instances/Individuals: The basic or "ground level" objects. An instance/individual is a real-world entity or an entity group. Normally, an instance is mapped to one or more concepts.

IS: Information System.

IT: Information Technology.

Knowledge Base: Ontologies and instances.

Knowledge Management: Refers to a range of practices and techniques used by organisations to identify, represent and distribute knowledge, know-how, expertise, intellectual capital and other forms of knowledge for leverage, re-use and transfer of knowledge and learning across the organisation.

Ontology: The collection of statements written in a language such as RDF that define the relations between concepts and specify logical rules for reasoning about them. Computers will "understand" the meaning of semantic data on a Webpage by following links to specified ontologies. Concepts (classes), properties (attributes), and relationships (relations) constitute an ontology. Most ontology describes concepts (classes), properties (attributes), relationships (relations), and instances (individuals).

Ontology Elements: Ontology comprises three types of elements: concepts, properties and relationships.

OWL: Web Ontology Language. It is a markup language for publishing and sharing data using ontologies on the Internet.

Property/Attributes: Features, characteristics, or parameters that objects can have and share.

RDF: Resource Description Framework, which is a scheme for defining information on the Web. RDF provides the technology for expressing the meaning of terms and concepts in a form that computers can readily process. RDF can use XML for its syntax and URIs to specify entities, concepts, properties and relations.

RDF Schema: RDF's vocabulary description language, which defines classes and properties that may be used to describe classes, properties and other resources. The main RDFS components are included in the more expressive language OWL.

Relationships/Relations: Ways that objects can be related to one another.

Resource: Web jargon for any entity. It refers to documents, people, materials, Webpages, parts of a Webpage, devices, and more.

RFI: Requests For Information

Semantic Web: The next generation Web containing action-able information (i.e. information derived from data through a semantic theory so that it can be processed directly and indirectly by machines). Tim Berners-Lee (2001) defined the Semantic Web as a Web, in which information is given well-defined meaning. Both computers and people can work in co-operation. In this thesis, the Semantic Web is regarded as an Information Technology which comprises the standards and tools of several techniques: XML, XML Schema, RDF, RDF Schema, OWL, etc.

SWIMS: Semantic Web-based Information Management System for construction projects.

Taxonomy: Refers to either a classification of things, or the principles underlying the classification. Almost anything, animate objects, inanimate objects, places, and events, may be classified according to some taxonomic scheme. Taxonomies are frequently hierarchical in structure, having parent/child relationships. However, taxonomy may also refer to relationship schemes other than hierarchies. It is slightly less broad than ontologies. In this thesis, a taxonomy consists of the terms used to define ontologies, while ontologies describe those interrelated terms.

URI: Universal Resource Identifier. A URI defines or specifies an entity, not necessarily by naming its location on the Web.

URL: Uniform Resource Locator. URLs are the most familiar type of URI.

W3C: World Wide Web Consortium. It is an international consortium where member organisations, full-time staff and the public work together to develop standards for the World Wide Web.

Web Services: A software system designed to support interoperable machine-to-machine interaction over a network. It is a set of tools that can be used in a number of ways, but no specification that defines it so far.

Web and **WWW**: World Wide Web.

XML: eXtensible Markup Language. A markup language lets individuals to define and use their own tags. XML has no built-in mechanism to convey the meaning of the user's new tags to other users.

XML Schema: A XML schema language, which uses a rich datatyping system, allowing for more detailed constraints on an XML document's logical structure, and is required to be processed in a more robust validation framework.

TABLE OF CONTENTS

ABSTRACT ..... *i*

ACKNOWLEDGEMENTS ..... *ii*

GLOSSARY ..... *iii*

CHAPTER 1 - INTRODUCTION *1*

1.1 Introduction 1

1.2 Research Background 1

1.3 Challenges and Opportunities 4

1.4 Aim and Objectives 7

1.5 Research Scope 7

1.6 Justification for the Research 9

1.7 Contributions 10

1.8 Thesis Layout and Contents 11

CHAPTER 2 - THE SEMANTIC WEB *15*

2.1 Introduction 15

2.2 The Semantic Web Concept and Technology 15

2.2.1 Definition 15

2.2.2 XML and XML Schema 19

2.2.3 RDF and RDF Schema 20

2.2.4 Ontology Vocabulary 24

2.2.5 Summary 26

2.3 Semantic Web Applications 27

2.3.1 Industrial Applications 27

(1) Knowledge Management 28

(2) Web Services 33

2.3.2 Potential Applications in Construction Industry 35

(1) Knowledge Management 39

(2) Collaborative Design 40
(3) On-line Procurement 41
(4) On Site Information Management 43
(5) Communication and Collaboration 44
(6) Change and Claim Management 45
2.3.3 The Key Features, Enablers, Barriers, and Potential Benefits from Adopting the Semantic Web 46
2.4 Semantic Web Software 47
2.4.1 RDF/XML Parser 47
2.4.2 Ontology Editors 48
(1) OilEd 48
(2) Ontolingua 49
(3) WebODE 51
(4) OpenCyc 51
(5) OntoEdit, OntoBuilder and OntoShare 52
(6) Protégé 53
2.4.3 OWL Reasoners 55
2.5 Summary 57
CHAPTER 3 - CONSTRUCTION PROJECT INFORMATION MANAGEMENT *58*
3.1 Introduction 58
3.2 Definition 58
3.2.1 Information Management 58
3.2.2 Construction Project Information 59
3.2.3 Information Management and Document Management 59
3.3 Historical Perspective 60
3.3.1 Paper-based Manual Information Management Stage 60
3.3.2 Electronic Information Management Stage 61
3.3.3 Web-based Information Management Stage 61
3.4 Challenges 62
3.5 Semantic Web Opportunities 64

3.6 Information in Construction Project Documents 65
3.6.1 Document Types 65
(1) From the Construction Project Stages in Which Documents are Used... 66
(2) From the Producers and Users of Documents 68
(3) From the Subject of Documents Content 70
(4) From the Format of Documents and Presentation Mediums 71
(5) Discussion 72
3.6.2 Construction Product Information Modelling Mechanism 72
(1) Standard for the Exchange of Product Model Data (STEP) 73
(2) Industry Foundation Classes 73
3.6.3 Construction Project Information Classification Systems and Taxonomies 75
(1) ISO Framework for Information Classification 75
(2) LexiCon and bcBuildingDefinition 79
(3) IFC 81
(4) UniClass 84
3.6.4 XML-based Languages for Construction Project Information Schema .. 87
3.6.5 Taxonomy in This Research 87
3.7 Summary 91
CHAPTER 4 - METHODOLOGY *92*
4.1 Introduction 92
4.2 Methodologies in Information Systems Research 92
4.2.1 Action Research 94
4.2.2 Case Studies 95
4.2.3 Surveys 96
4.2.4 Laboratory and Field Experiments 98
4.2.5 Futures Research 98
4.2.6 Longitudinal Studies 100
4.2.7 Simulations 100
4.2.8 Phenomenological Studies 101
4.2.9 Ethnographic Studies 101
4.3 Research Process and Methodology Adopted 101

4.3.1 Understanding the Research Question 101
4.3.2 Research Steps and Methods 103
Step One: Identify available Semantic Web techniques and tools 105
Step Two: Identify characteristics of construction project information 105
Step Three: Identify existing construction information management tools and review existing construction information classification systems 105
Step Four: Identify potential scenarios 106
Step Five: Develop and revise framework 108
Step Six: Develop and revise components and implementation mechanisms 108
Step Seven: Deploy the system in practical project information management 109
Step Eight: Evaluate the system framework and deployment 110
4.3.3 Justification of Research Methods Adopted 111
4.4 Summary 113
CHAPTER 5 - SWIMS FRAMEWORK, APPLICATION SCENARIOS AND ANNOTATIONS 114
5.1 Introduction 114
5.2 Framework 114
5.2.1 Overview 114
(1) Human Interpretable Information Layer 115
(2) Computer Interpretable Information Layer 116
5.2.2 Personnel and Data Repository 117
5.2.3 Interface Application 117
5.2.4 Annotated Data Repository 118
5.2.5 Intelligent Agents 118
5.3 Scenarios Deployment 119
5.3.1 Scenario 1 - Document Management Application 119
(1) Annotation of Documents 120
(2) Annotation of Project Partners 120
(3) Document Update Alerting Service 121

(4) Semantic Web-based Query 121
(5) Precise Response 121
(6) Report Generator 121
5.3.2 Scenario 2 - Product Specification Conversion Application 122
5.3.3 Scenario 3 - Terminology Merging Application 128
5.4 Annotations and Metafiles 129
5.4.1 Annotation of Documents in Other Industry Sectors 129
5.4.2 Annotation of Construction Project Documents 134
(1) Metafile Model 134
(2) Items of the Metafile 134
5.4.3 Annotation of Project Partners 141
(1) User Profile Model 141
(2) Items of User Profile 142
5.5 Summary 143
CHAPTER 6 - ONTOLOGY DEVELOPMENT FOR SWIMS 145
6.1 Introduction 145
6.2 Ontology Development Methodologies 145
6.3 Tools in This Research 152
6.4 Functions and Components of SWIMS Ontology 153
6.4.1 Functions 153
6.4.2 Components 153
6.5 SWIMS Ontology and Instances Definitions 154
Step One: Determine the domain and scope of the ontology 154
Step Two: Consider reusing existing ontologies 156
Step Three: Enumerate important terms in the ontology 159
Step Four: Define the concepts' classes and arrange the class in a taxonomic hierarchy (subclass- upperclass) 162
Step Five: Define the properties of concepts 167
Step Six: Define the allowed values for properties 172
Step Seven: Create instances to fill in the value for properties 173
6.6 Shared Ontology and Speciality Ontology 177

6.7 Ontology Validation 178

6.8 Ontology Management 179

6.9 Summary 182

CHAPTER 7 - IMPLEMENTATION OF SWIMS DEMONSTRATOR 184

7.1 Introduction 184

7.2 Implementation Environment 184

7.2.1 Hardware 184

(1) From the End-user's Perspective 184

(2) From the Ontology Engineer's Perspective 184

7.2.2 Software 185

(1) From the End-user's Perspective 185

(2) From the Ontology Engineer's Perspective 185

7.3 Prototype and Demonstrator 187

7.3.1 Management of Ontology 189

(1) Develop Ontology 190

(2) Merge Ontology 193

(3) Update Ontology 196

(4) Remove Ontology 197

7.3.2 Management of Actor Information 197

(1) Creation of User Profile 198

(2) Management, Updating and Removal of User Profiles 199

7.3.3 Management of Documented Information 201

(1) Document Upload 202

(2) Document Management 204

(3) Report Generation 224

(4) Remove Document 224

7.4 Summary 224

7.4.1 Conclusion of the Demonstrator 224

7.4.2 The Main Functions and Benefits of Adopting SWIMS 225

(1) Functions 225

(2) Benefits 226

7.4.3 IT Tools Used 226

(1) Software 226
(2) Hardware 227

CHAPTER 8 - SYSTEM EVALUATION *228*

8.1 Introduction 228

8.2 Evaluation Objectives 228

8.3 Methodology 230

8.3.1 Select Evaluation Techniques 230

8.3.2 Choice of Evaluators 232

8.3.3 Design of Evaluation Demos 233

8.3.4 Design of Evaluation Questionnaire 233

8.4 Evaluation of SWIMS Framework and Deployment Examples 234

8.4.1 Evaluation Process 234

8.4.2 Analysis of Results 235

(1) Evaluators' Background 235

(2) Responses to Questions about SWIMS 237

(3) Responses to Questions about the Effectiveness of the Evaluation Approach 248

8.5 Discussion and Summary 248

CHAPTER 9- CONCLUSIONS AND FURTHER DEVELOPMENTS *251*

9.1 Introduction 251

9.2 Summary 251

9.3 Conclusions 256

9.4 The Limitations of This Research 257

9.5 Recommendations for Further Work 258

9.5.1 Recommendations for Further Research 258

(1) Recommendations for Refining SWIMS 258

(2) Recommendations for Developing the Semantic Web and Its Tools....261

9.5.2 Recommendations for the Construction Industry 262

9.6 Closing Remarks 263

REFERENCE 265

APPENDIX A -PUBLICATIONS.....A-1

APPENDIX B -CONCEPT HIERARCHY OF SWIMS SHARED ONTOLOGY 1.0 .....B-1

APPENDIX C -CONCEPT HIERARCHY OF SWIMS PRODUCT ONTOLOGY 1.0.....C-1

APPENDIX D -CONCEPT HIERARCHY OF SWIMS CLIENT ONTOLOGY 1.0 .....D-1

APPENDIX E -QUESTIONNAIRE E-1

LIST OF FIGURES

Figure 1-1: Research Scope 8
Figure 2-1: The Semantic Web Tower (Berners-Lee, 2002) 16
Figure 2-2: A Key Loop in the Semantic Web (Berners-Lee, 1998) 17
Figure 2-3: RDF Form 21
Figure 2-4: Semantic Links of RDF Data 22
Figure 2-5: OntoWeb Search Interface (OntoWeb, 2003) 31
Figure 2-6: Toolset Environment of On-To-Knowledge (On-To-Knowledge, 2002) 32
Figure 2-7: The Architecture of PlanetOnto (Domingue and Motta, 2000) 35
Figure 2-8: Interface of Search on e-COGNOS Server (e-COGNOS Consortium, 2002) 38
Figure 2-9: Interface of Search Result on e-COGNOS Server (e-COGNOS Consortium, 2002) 39
Figure 2-10: Construction Knowledge Management Using the Semantic Web (Anumba et al, 2003) 40
Figure 2-11: Properties of Objects Material and Concrete, Generated by the Taxonomy Server (bcXML, 2003) 42
Figure 2-12: Deployment Architecture of Context-aware Services for Mobile Computing in the Construction Industry (Aziz et al, 2006) 44
Figure 2-13: OilEd Ontology Editor Example (Bechhofer et al, 2001) 49
Figure 2-14: Class Editing in Ontolingua (Farquhar et al, 1997) 50
Figure 2-15: Class Definition Browser in Ontolingua (Farquhar et al, 1997) 50
Figure 2-16: WebODE Ontology Editor (Gómez-Pérez et al, 2003) 51
Figure 2-17: OpenCyc KB Browser (OpenCyc.org, 2005) 52
Figure 2-18: Protégé-OWL Screenshot - OWLClasses View (Protégé, 2006) 54
Figure 2-19: Protégé-OWL Screenshot- Visualised OWL Ontology (Protégé, 2006) 55
Figure 3-1: A Mass of Documents in a Construction Project 60
Figure 3-2: Fragmentation during Project Phases and among Partners (Emmerik, 2000) 69

Figure 3-3: Project Documents and Project Partners 70
Figure 3-4: IFC2x3 Architecture (IAI, 2006 b) 74
Figure 3-5: Classes and the General Relationships between Classes (ISO, 2001) ... 76
Figure 3-6: Construction Process Model (ISO, 2001) 77
Figure 3-7: Entities of the Model in Standard ISO/DIS 12006-3 78
Figure 3-8: Objects Schema Breakdown by LexiCon 79
Figure 3-9: Object Hierarchy by the LexiCon Taxonomy Server (STABU, 2004) .. 80
Figure 3-10: IFC Entity Schema Breakdown 82
Figure 3-11: UniClass Table Lists 85
Figure 4-1: A Multi-methodological Approach to IS Research (Nunamaker and Chen, 1990-91) 93
Figure 4-2: RDF of Research Question 102
Figure 4-3: Research Steps and Methods Adopted 104
Figure 5-1: Framework of Traditional Information Management Systems 115
Figure 5-2: SWIMS Framework 116
Figure 5-3: Scenario 1 - Document Management 120
Figure 5-4: Scenario 2a - Measurement System Conversion 123
Figure 5-5: Scenario 2b - Technical Standard Conversion 127
Figure 5-6: Standard Conversion Ontology Represented in RDF(S) and RDF 128
Figure 5-7: Scenario 3 - Terminology Merging 129
Figure 5-8: Annotations on a Radio Broadcast (Dowman et al, 2005) 130
Figure 5-9: Index of an Annotated Document (Dowman et al, 2005) 130
Figure 5-10: Listing Annotated Documents in OntoShare (Davies et al, 2003) 131
Figure 5-11: Annotating Ontological Keywords to a Document in OntoShare (Davies et al, 2003) 132
Figure 5-12: Identifying Expertise in OntoShare (Davies et al, 2003) 132
Figure 5-13: Document Annotation on e-COGNOS Server (e-COGNOS Consortium, 2002) 134
Figure 5-14: Document Metafile 135
Figure 5-15: An Example of a Functional Property: Last Accessed By 139
Figure 5-16: An Example of a Transitive Property: Refer To 140
Figure 5-17: An Example of an Inverse Property: "Include Document" has an inverse property that is "Belong To" 140
Figure 5-18: An Example of a Symmetric Property: In Conjunction With 141

Figure 5-19: User Profile Model 142
Figure 6-1: OTK Knowledge Management Application Development Process (On-To-Knowledge, 2002) 151
Figure 6-2: The Mapping of SWIMS Framework to IFC Model 158
Figure 6-3: Annotation Property of a Concept - actor 170
Figure 6-4: Annotation Property of an Instance - a building service engineer 170
Figure 6-5: Annotation Property of a Datatype Property - *DateProduced* 170
Figure 6-6: Annotation Property of an Object Property - *ProduceDocument* 170
Figure 6-7: Screenshot of Document Class in Protégé 174
Figure 6-8: Screenshot of a Metafile for a Project Document in Protégé 175
Figure 6-9: Properties of an Organisation Actor 175
Figure 6-10: Screenshot of a Metafile for an Organisation in Protégé 176
Figure 6-11: Prompt Ontology Management Framework (Noy and Musen, 2004) 182
Figure 7-1: An IDEF0 Node ((Hanrahan, 1995)) 186
Figure 7-2: Hierarchical View of IDEF0 Model (Hanrahan, 1995) 187
Figure 7-3: Construction Project Information Management Process in IDEF0- Node A-0 188
Figure 7-4: Construction Project Information Management Process in IDEF0- Node A0 189
Figure 7-5: Ontology Management Process in IDEF0- Node A1 190
Figure 7-6 : Screenshot of SWIMS Shared Ontology 1.0 in Protégé 192
Figure 7-7: Screenshot of SWIMS Product Ontology 1.0 in Protégé 192
Figure 7-8: Screenshot of SWIMS Document Speciality Ontology 1.0 in Protégé .. 193
Figure 7-9: Screenshot of the Merging Shared Ontology with Product Ontology . 195
Figure 7-10: Merged SWIMS Ontology 195
Figure 7-11: Check Consistency and Update Ontology in Protégé 196
Figure 7-12: Actor Information Management Process in IDEF0- Node A2 197
Figure 7-13: Screenshot of People Class and Properties in Protégé 198
Figure 7-14: User Profile of Architect in Protégé 199
Figure 7-15: Edit Slots of User Profile 200
Figure 7-16: Documented Information Management Process in IDEF0- Node A3 201
Figure 7-17: Document Upload Process in IDEF0- Node A4 202
Figure 7-18: Metafile of a Project Document 203

Figure 7-19: Document Management Process in IDEF0- Node A5 204
Figure 7-20: Relationship between Document, Concept and Keywords 206
Figure 7-21: Client's Document Repository before Classification.....209
Figure 7-22: Definition of the Technical Document.....209
Figure 7-23: Classified Documents into the Client's Category 210
Figure 7-24: Definition of the Manufacturer Drawing 211
Figure 7-25: Classified Project Manager's Category.....212
Figure 7-26: Project Manager's Document Repository before Classification.....212
Figure 7-27: Categorising Documents into the Project Manager's Category.....213
Figure 7-28: Compare Terminologies of Architect and Structural Engineer 215
Figure 7-29: Aggregated Terminologies before Mapping 215
Figure 7-30: Concept Hierarchy View with Unmapped Terminologies.....216
Figure 7-31: Mapping Terminologies and Merging Ontologies in Protégé 217
Figure 7-32: Visualise the Merged Ontology 218
Figure 7-33: Search Documents Based on Their Semantics-Example 1a 220
Figure 7-34: Search Documents Based on Their Semantics-Example 1b 220
Figure 7-35: Search Documents Based on Their Semantics-Example 1c 220
Figure 7-36: Search Documents Based on Their Semantics-Example 2a 221
Figure 7-37: Search Documents Based on Their Semantics-Example 2b 222
Figure 7-38: Search Documents Based on Their Semantics-Example 2c 222
Figure 7-39: Search Documents Based on Their Semantics-Example 2d 222
Figure 7-40: End-user Interface Template of SWIMS Document Search Engine....223

LIST OF TABLES

Table 2-1: W3C Recommendations for the Semantic Web 19
Table 2-2: RDF Parsers Review List (McBride et al, 2003) 48
Table 3-1: Common Process and Documents of a Construction Project (Process Protocol, 2002) 67
Table 3-2: Exemplary Project Document Classification by Subject Content 71
Table 3-3: General Electronic Project Document Presentation Medium and Format 72
Table 3-4: Index of UniClass Tables (CPIC, 1997) 86
Table 4-1: Definitions of Research Approaches Used in IS Research 94
Table 4-2: Comparison between Open and Closed Questions (Foddy, 1994) 97
Table 4-3: Distinctions between Quantitative and Qualitative Data (Saunders et al, 2003) 97
Table 5-1: Cable Suppliers List 123
Table 6-1: Project Partner-related Terms 159
Table 6-2: Building-related Terms 160
Table 8-1: Evaluators' Industry Background 235
Table 8-2: Evaluators' Related Work Experience 236
Table 8-3: Evaluators' Position in Their Organisations (excluding the IT expert) 236
Table 8-4: Evaluator's IT Background 236
Table 8-5: Responses to Feasibility and Compatibility of the SWIMS 238
Table 8-6: Responses to the Sustainability of SWIMS and Efficiency of Use 238
Table 8-7: Responses to the Simplicity of Using SWIMS 240
Table 8-8: Responses to the Rationality of the SWIMS Deployment Example 241
Table 8-9: Responses to the Suitability of SWIMS 244

LIST OF LISTINGS

Listing 2-1: XML Syntax 19
Listing 2-2: A Project Document in XML 20
Listing 2-3: RDF Triple Model 20
Listing 2-4: RDF Schema Example - Class/Concept Definitions 23
Listing 2-5: RDF Schema Example - Carpenter Group Members (Instances) 24
Listing 2-6: Ontology for Building 25
Listing 5-1: RDF Annotation for Cable Case 124
Listing 5-2: RDF Annotation for Cable Supplier 125
Listing 5-3: Partial Ontology for Cable Case 126
Listing 6-1: Dataype Property Set on SWIMS Ontology 168
Listing 6-2: Object Property Set on SWIMS Ontology 169

LIST OF EQUATIONS

Equation 5-1 136
Equation 5-2 137
Equation 5-3 137
Equation 5-4 138
Equation 5-5 138

CHAPTER 1 - INTRODUCTION

1.1 INTRODUCTION

The Semantic Web is an emerging technology which provides intelligent access to heterogeneous distributed information, enabling software applications to mediate between user needs and information sources (Berners-Lee et al, 2001 and Fensel, 2001). The work described in this thesis reflects an effort in exploring the Semantic Web's ability to support information management in construction projects. By establishing a framework of a Semantic Web-based information management system and prototyping the system, the semantic management of construction information was explored, resulting in some implementation demos. This contributes to the integration of heterogeneous construction information, software applications and ontologies for the collaboration between project team members.

This chapter consists of a brief introduction background to the research, specific problems being addressed, aims and objectives, research scope, summary of contributions and the structure of the thesis.

1.2 RESEARCH BACKGROUND

Construction projects include a series of complex tasks involving intensive information from project partners (clients, architects, structural engineers, building service designers, and manufacturers to name a few). Modern construction projects, characterised by close co-operation among their globally distributed partners, require accurate and timely sharing of information between multiple disciplines and interoperability between independent databases (Aziz et al, 2004). Each partner manipulates volumes of documents which are normally in diverse formats. Computer-aided management systems are being used by Architectural, Engineering and Construction (AEC) companies to facilitate their activities and enhance their ability to process large amount of information in a relatively short space of time. Many applications, databases and management systems have been used to facilitate information management within a company. For instance, an electronic information management system can help project partners handle information digitally, which reduces the space for document storage, saves time for document delivery to

multiple offices and enables document retrieval by search engines. Since project partners are often multi-disciplinary and may be geographically distributed, efficient information exchange among project partners is a vital issue in achieving project collaboration (Faraj and Alshawi, 2004). Various methods have been used in supporting the share of project information, such as face-to-face meetings, postal correspondence, and tele-conferences. Furthermore, with the widespread use of the World Wide Web, Web-based collaborative working systems/tools have been developed to facilitate information exchange (van Leeuwen and van der Zee, 2005), such as extranet document management platforms. Web-enabled electronic information management systems enlarge the reachable scope of information and enable global co-operation among partners.

Although considerable research effort and practice have gone into how to manipulate project information within a company and deliver documents between offices, the current Web-based information management approach has the following limitations:

(1) Lack of content-based information management

Current Web-based information management systems do not reach their full potential because the Internet is often only used for display purposes, such that data can only be shared and processed by humans. With more and more accessible information, there is an overload on users. It requires users to manually manipulate information such as converting technical terminologies and filtering out imprecise search responses. It is inconvenient to solve problems when project partners cannot meet together at the same time, use different languages, or have different perspectives on an issue. This often leads to low efficiency and mistakes in communicating project information and in collaboration among project partners, which are major hurdles for the success of projects.

(2) Lack of interoperability of information sources

When more and more information sources and applications are connected via the World Wide Web, the interoperation and integration issues gain more attention by project partners. An investigation by the National Institute of Standards and Technology (NIST) showed that inefficient interoperability costs the U.S.

construction industry more than $15.8 billion on the design, construction and maintenance of large commercial, institutional and industrial buildings in 2002. This number is even higher if public works and other civil infrastructures systems are considered (Gallaher et al, 2004).

(3) Lack of context-awareness for Web Services

Construction project delivery involves a dynamic process. The information needs of a project partner depend on his/her context (role, time and location). Unfortunately, current Web-based information management tools cannot provide context-aware services. This results in considerable information overload during distribution and retrieval processes.

One of the approaches that attempts to solve the above three problems is to universally translate information among various project partners. Information and Requests For Information (RFI) are defined in such a way that they can be automatically 'understood' and processed by computers. The term, "Semantic Web", or "Web of Meaning", is used to describe a Web, in which information is given well-defined meaning, such that both computers and people can work in co-operation (Berners-Lee et al, 2001). Since the information on the Semantic Web has a clearly defined meaning, it can be analysed and traced by computer programs. Although information management tools (software/applications/databases/systems) on the Semantic Web may be designed independently, they will be able to share and process data intelligently.

The Semantic Web groups and classifies information into concepts and recognises the relationships between concepts, information sources and users. It enables inter-system information exchange to be processed in a logical, intelligent and efficient way. Therefore, it is promising to enhance the quality of activities requiring accurate and timely information exchange, such as collaborative design, procurement, and project information retrieval. It is expected that the Semantic Web could be applied in the construction industry to facilitate the building and construction process, and overcome the major shortcomings of existing information management systems.

1.3 CHALLENGES AND OPPORTUNITIES

Existing information management technologies support information communication to a reasonable extent. However, they still require users to manually manipulate information in many circumstances, such as negotiating with working groups about the new plan due to design changes. Therefore, it is inconvenient to resolve problems and reach an agreement when project partners cannot meet together, use different languages, standards or terminologies. Most existing electronic document management systems use keyword-based search engines. The keywords of a document are often pre-defined. In a query, all and only the documents, whose keyword matches the searched word exactly will be listed. However this type of search does not address synonym and multivocal issues, which leads to information loss and possible overload. Meanwhile, the interoperability of different project management tools (software and applications) is hardly addressed. Most times, partners have to recognise the output from application A and manually reinput it into application B because the applications may be programmed using different languages and data structures. At the same time, it is hard to maintain the consistency of updated information throughout a variety of applications. Context-awareness is another issue which has rarely been addressed by current information management systems. These lead to inefficiencies in managing project information among partners. More efficient information management is expected to be the primary mechanism for the construction industry to increase its productivity (Egan, 1998). Therefore, a solution is required to enhance electronic information management, which can be applied not only within individual companies, but also across all the project partners, and to process information on a content and context-aware basis.

The Semantic Web provides many good opportunities to meet the above challenges. Some important ways in which the Semantic Web can improve the status quo are briefly described below:

- Automatic information processing on the Web: Computers can reach agreement on an issue following the information given by project partners, who cannot sit together for face to face negotiations.

- Advanced content-based and context-specific information management: Taking the information retrieval as an example, in a query on the Semantic Web, computers can identify the context of a user and the content of RFI, infer the semantic relations of individuals to decide the scope of the search, apply context transformation, use domain-specific rules to reason over concepts of the ontology, and filter out facts that are irrelevant to the answer. Consequently, users receive more flexible and precise responses.

- Enhanced knowledge management, with timely updating and sharing of information: Ontologies are a key technology, at the semantic level, to share information between information resources when they use different conceptualisations, especially in decentralised and dynamic environments (Stuckenschmidt and van Harmelen, 2004), such as in construction projects. Whenever changes occur, computers can make logic reasoning, identify related documents/database and update information/data in it, and identify the most appropriate project participants and share the updated information/data with them. Each ontology is edited for a knowledge management task, used as a plug-in to a construction information management system, and can be reused in similar cases.

- Reduction of information overload: On the Semantic Web, resources, all things containing information, are annotated by a set of attributes. Information can be sent to the user only when the "is About" attribute of information matches the "Interested In" attribute of the user. This could be considered as a push mode of information dissemination activity. There is also a pull mode of information dissemination on the Semantic Web, in which users get the most relevant responses to their queries by applying ontology-based information retrieval. Therefore, information dissemination on the Semantic Web is a need-based service, with less redundancy and reduced load of irrelevant information.

- Interoperation between independently developed applications: Once the Semantic Web is built, with the automatic operation by computers, heterogeneous information can be recognised by all software applications. Information Technology will reach its full potential in project information management.

However, to transfer construction project information management to a Semantic Web environment, many challenges exist:

- No systematic information management framework on the Semantic Web basis.

One of the most serious challenges is the need to establish an information system which employs the Semantic Web technology to construction information management domain. Within the system framework, diverse databases, ontologies, applications (or whatever) can co-operate. To date there is no research which has systematically addressed this issue.

- Lack of information annotation prototype.

Another big challenge is the need to annotate information, relate it by ontology and process it. There is no prototype annotation and no standard to specify what attributes are necessary to be annotated.

- Lack of semantic content in construction project documents and poorly structured data/information.

Documents are a major medium for project information storage and sharing. Because of fragmentation and diversity of construction information, most construction project documents lack adequate semantic content for manipulation in a Semantic Web environment. They need to be enriched to make the information explicit to both humans and machines and easy to be retrieved. Meanwhile, many documents are often poorly structured, which makes information retrieval very difficult. Mechanisms to annotate semi-structured or un-structured documents are beneficial to allow computers to process the documents based on their content.

- No common ontology and ontology library for construction:

At the moment, there is no universally agreed ontology for construction concepts and terms. How to translate the complex relationships between individuals in the real world into hierarchical classes of ontologies on the Semantic Web needs to be studied. Methods for collecting various ontologies and merging related ones are under development.

For construction project information management, the challenges highlighted above may be accentuated by the fragmented nature of the construction industry.

Interestingly, this also represents tremendous opportunities for this industry to gain considerable benefits from improvement of current situation. This thesis presents a Semantic Web-based construction project Information Management System (SWIMS) that is intended to contribute the body of knowledge in this regard.

1.4 AIM AND OBJECTIVES

The research project aims to investigate how construction project information can be managed within a Semantic Web environment. In achieving this goal, the following objectives were defined:

(1) To review Semantic Web theories and background technologies.

(2) To review implementations of the Semantic Web technologies in other industries in order to identify the key features, enablers, barriers, and potential benefits.

(3) To explore the range of construction project information and identify how these can best be managed in a Semantic Web environment.

(4) To implement a system framework and its middleware for managing construction project information in a Semantic Web environment.

(5) To implement prototype system using practical project information and evaluate the system demonstrator.

1.5 RESEARCH SCOPE

This research mainly focused on exploring the Semantic Web technology in construction information management to improve efficient and accurate sharing of documented information.

A traditional Web-based information management environment consists of three parts: hardware, software/applications and human beings/information sources. A Semantic Web-based information management environment involves all the above three parts as well as the Semantic Web technology, which consists of several techniques. The Semantic Web technology can be considered as middleware in the environment, as depicted in Figure 1-1. Information developed by Semantic Web techniques is processable by computers. Using the developed information,

software/applications could provide precise information to human beings in a timely fashion.

This research was focused on construction information (data sources) and its management using Semantic Web techniques. It consists of the representations of construction information and the processes of information management. The human factors, management techniques, hardware and software/applications were not considered at this time.

This research worked on the semantic components (middleware), which employed specific Semantic Web techniques for information management. Information models were proposed for these components, such as the annotation models (metafiles and user profiles), associated mechanisms (interactions between annotation items) and ontology of the managed information. Practical project information was operated on these models to simulate the processes of Semantic Web-based information management in real-world projects. Potential application areas and scenarios were also proposed.

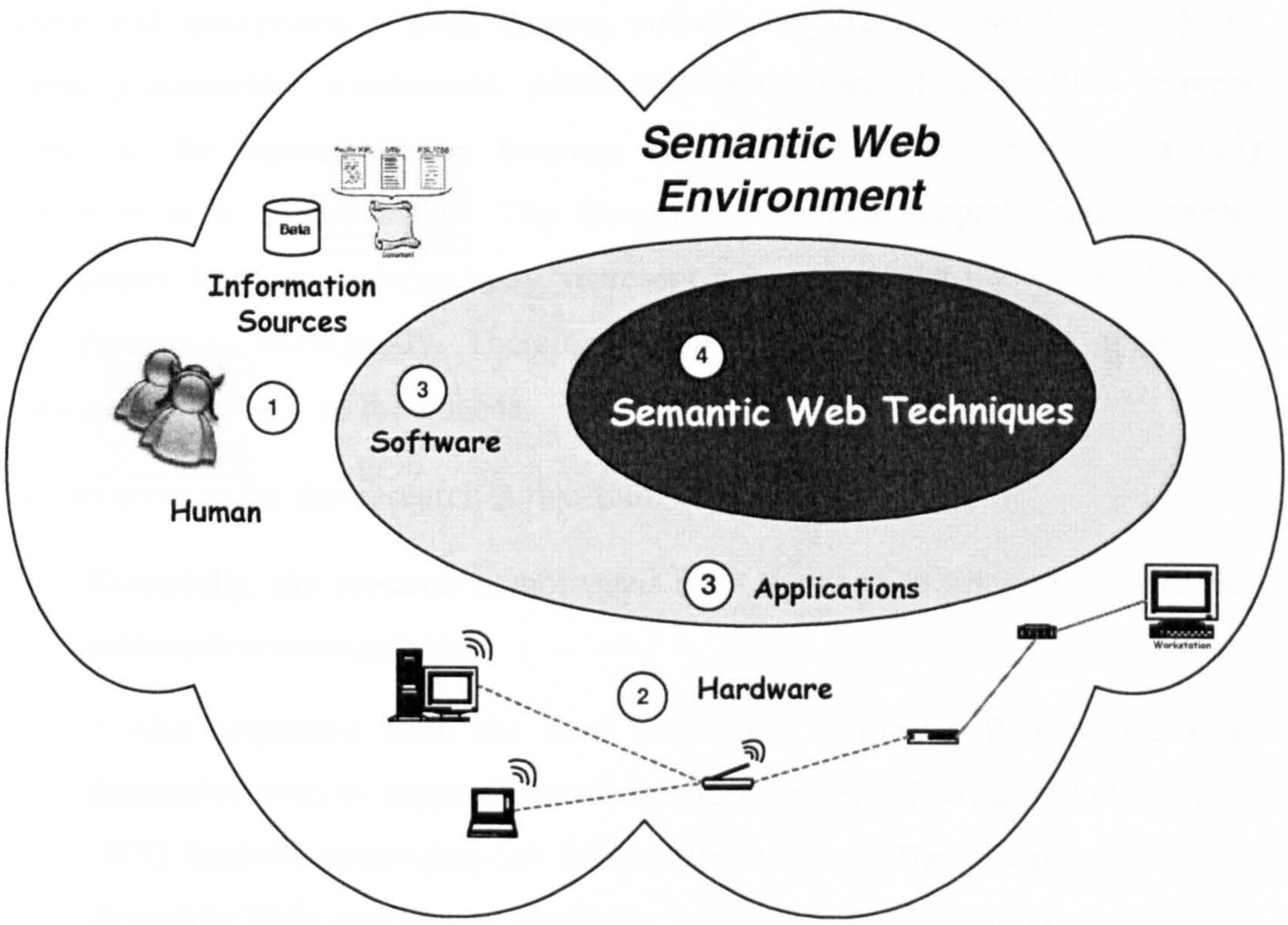


Figure 1-1: Research Scope

In summary, this research developed middleware models within an IS model, enabling the system to support various end-user software/applications to manage heterogeneous construction information from diverse project partners, in the Semantic Web environment.

1.6 JUSTIFICATION FOR THE RESEARCH

The construction industry is fragmented from both geographical and functional perspectives. Geographically, project partners may be distributed with constant remote collaboration. Functionally, the entire construction process may involve partners from a variety of disciplines. Information needs change with the changing context (role, time, location) which the project partner is in. The fragmentation is a challenge to the traditional construction information management practices because most project team members use discrete software package to manage the construction information while the information exchange between the team members is done through a time consuming manual process using paper-based media (Sun et al, 2003). The advent of the World Wide Web brought the distributed information sources (human, software/applications and databases) to a virtually connected environment, which reduces the time of information delivery. However, the interoperability between diverse information sources was only improved to a limited extent. The Semantic Web can integrate heterogeneous information into a knowledge body, represent it in a computer-understandable way and process it intelligently. Therefore, project partners can access appropriate information on basis of their needs.

The motivation for this research is two-fold:

(1) Essentially, the research is motivated from the need to improve construction information management.

(2) It also originated from the need to develop Semantic Web as the next-generation Web to support Knowledge Management (KM) and Web Services (WS) between heterogeneous information sources. The functionality of the Semantic Web provides a promising approach to solve current information management problems derived from the fragmented nature of the construction industry (Pan et al, 2003). The preliminary applications of the Semantic Web technology in other industries provide valuable experience for this research.

Meanwhile, since the Semantic Web is still in its infancy, the research also provides some experiences for the development of the Semantic Web technology.

1.7 CONTRIBUTIONS

The fragmented nature of the construction industry leads to difficulties in managing construction information among project partners and providing information to specific participants appropriately. Although different information technologies have been applied, none of them, thus far, completely resolve the above problems. The development of the Semantic Web provides an innovative approach which enables the inter-operation between diverse information sources and supports context-aware information services. By adopting this technology, construction information can be identified and processed by computers as well as by human beings, so as to improve the efficiency of information management. This can hardly be achieved by other technologies. Current studies related to the applications of Semantic Web technology in the construction sector are few and fragmented. Most of them focused on the development of ontologies for specific construction domains and lack a systematic view on the Semantic Web-based information management. The specific contributions of this research can be summarised as follows:

(1) This research analysed the features of construction information and pointed out problems underlying existing information management methods/tools. The Semantic Web technology was proposed as a possible solution to those problems.

(2) A Semantic Web-based Information Management System (Swims) framework was developed. It enables the representation of heterogeneous information by ontological concepts, merges fragmented existing ontologies, extensively incorporates new ontologies, integrates ontologies for completing various construction management tasks, and has the potential to provide various content-based and context-aware services to project partners. Potential system application areas and scenarios were identified.

(3) An information annotation model was developed. The model can be applied to annotate the semantics of various information resources, such as project documents, project partners, structural elements, and materials. The document annotation model (metafile) and project partner annotation model (user profile)

were addressed in detail in the thesis. This research identified semantic content and context of project documents and partners within a documented information management scenario. These semantics were represented in corresponding annotations.

(4) A shared ontology about generic construction information was developed and several lightweight speciality ontologies were developed within the documented information management domain for various project partners. These ontologies are compliant with the widely accepted Industry Foundation Classes (IFC). The compliance with IFC model enables the developed ontologies to be easily mapped to other IFC-based ontologies and to interact with IFC-based applications.

(5) This research designed a documented information management process and simulated the process by implementing Swims ontologies on practical project documents. The involved content-based and context-aware functions can not be achieved by traditional Web technologies.

Detailed descriptions of the SWIMS framework, annotation model, ontologies, and implementation are included in the thesis.

1.8 THESIS LAYOUT AND CONTENTS

The thesis consists of nine chapters. The content of each chapter is described below:

Chapter 1: Introduction

This chapter provides the general background to the research by identifying the fragmented nature of construction industry and highlighting the challenges and opportunities existing in current construction information management. The possible solutions provided by the Semantic Web are discussed. Then the research aims and objective, scope, justification and contributions are outlined. The final section presents the thesis layout and a brief summary of each chapter.

Chapter 2: The Semantic Web

This chapter explains the Semantic Web theories and techniques. Representative Semantic Web implementation projects in other industries are reviewed and

analysed. The key features, benefits, and potential application areas in the construction industry are highlighted.

Chapter 3: Construction Project Information Management

This chapter explores the history of information management systems and reviews current construction information management systems. Then the types of construction information are identified. Existing information classification systems/taxonomies are reviewed and compared. The Industry Foundation Classes (IFC) model is selected to establish the basic taxonomy of SWIMS Ontology in sequent chapters.

Chapter 4: Methodology

This chapter provides an introduction to general Information System research methodologies. Based on the understanding of research topic, eight steps are specified to achieve five research objectives. The specific methodologies adopted in each research step are identified and justified in detail.

Chapter 5: SWIMS Framework, Application Scenarios and Annotations

This chapter presents the framework for the Semantic Web-based Information Management System (SWIMS) for construction projects. The framework aligns general information system components with the Semantic Web architecture. The use of the system is illustrated in three future scenarios. These scenarios describe various distinguished features of SWIMS in the documented information management context.

This chapter also analyses typical semantic annotation examples and then illustrates the annotation of construction information resources, documents and users in particular, in tabular models. Items of the annotations are developed in detail, including the interaction mechanism of document's keywords, "is about" property, and concepts in the relevant ontology. The annotations involve important attributes of resources and relationships between resources, so that computers can identify the content and context of a resource and then process it properly. These are useful in managing documented construction information within SWIMS.

Chapter 6: Ontology Development for SWIMS

In this chapter eight existing ontology development methodologies are compared. Noy's methodologies (Noy and McGuinness, 2001; Noy and Musen, 2003) are selected to develop the first version of the SWIMS knowledge base (ontologies and instances) in seven steps and manage involved ontologies. The concept hierarchy, property set and relationships between concepts and properties constitute an ontology. Seven top-level concepts, three types of properties, and four relationships are specified in SWIMS Ontology 1.0. These ontology elements are used in the developed annotation models for further implementations.

Chapter 7: Implementation of SWIMS Demonstrator

The execution process of the SWIMS model in documented information management scenarios is the focus of this chapter. A number of system functions are described in three function modules including: ontology management, actor information management, and documented information management. The detailed functions of each module are presented, such as the ontology development, merge, update, and remove; user profiles creation, management, update and remove; documents upload, intelligent update, automatically classifying, knowledge-based searches and attribute-based searches. These function modules can provide appropriate information to interface applications enabling content-based and context-specific information management. A demonstrator is presented showing the major operations of the system.

Chapter 8: System Evaluation

This chapter describes the system evaluation process and presents the evaluation results. Firstly, five criteria for evaluating SWIMS framework/components, two criteria for evaluating the performance of SWIMS deployment examples, and three criteria for evaluating the prospective industry applications of SWIMS are explained. A Web-based multi-media evaluation methodology is designed based on the research characteristics. The evaluation results are analysed and some conclusions are drawn.

Chapter 9: Conclusion and Further Developments

This chapter provides a summary of the research and presents its conclusions. Recommendations for future research and the limitations of the presented systems are also included.

CHAPTER 2 - THE SEMANTIC WEB

2.1 INTRODUCTION

This chapter describes the concepts and applications of the Semantic Web. A layered architecture of the Semantic Web is presented first. The specific techniques, which constitute the Semantic Web, are then introduced. The applications of the Semantic Web in various industries are reviewed from two general fields, the Knowledge Management and the Web Services, with the demonstration of representative projects. Its potential applications in the construction industry are highlighted in particular. Finally, available software tools for the Semantic Web are briefly reviewed.

2.2 THE SEMANTIC WEB CONCEPT AND TECHNOLOGY

The current World Wide Web greatly facilitates information sharing. However, it cannot reach its full potential if it is only understood by human beings. One of the approaches that seeks to solve this problem lies in making the Web understandable to both machines and humans. The term, "Semantic Web", or "Web of Meaning", is used to describe such a Web, in which information is given well-defined meaning, such that computers and people can work in co-operation (Berners-Lee et al, 2001).

Although the initial idea of a machine-understandable Web appeared in 1989, its official name, The Semantic Web, was first used in 2001. Currently, numerous research projects are being carried out on the Semantic Web. Many applications are being developed in the fields of bio-informatics, e-Commerce, library science, and knowledge management (see Section 2.3.1). They aim to "bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users" (Berners-Lee et al, 2001) .

2.2.1 Definition

Tim Berners-Lee (2003), inventor of the World Wide Web and the Semantic Web, defined the Semantic Web as: "a web of machine-readable information, whose meaning is well-defined by standards: it absolutely needs the interoperable

infrastructure that only global standard protocols can provide". To achieve the Semantic Web, Web resources should be described in a way that makes their meaning explicit. Tim Berners-Lee outlined his vision for the Semantic Web as a layered architecture (Figure 2-1).

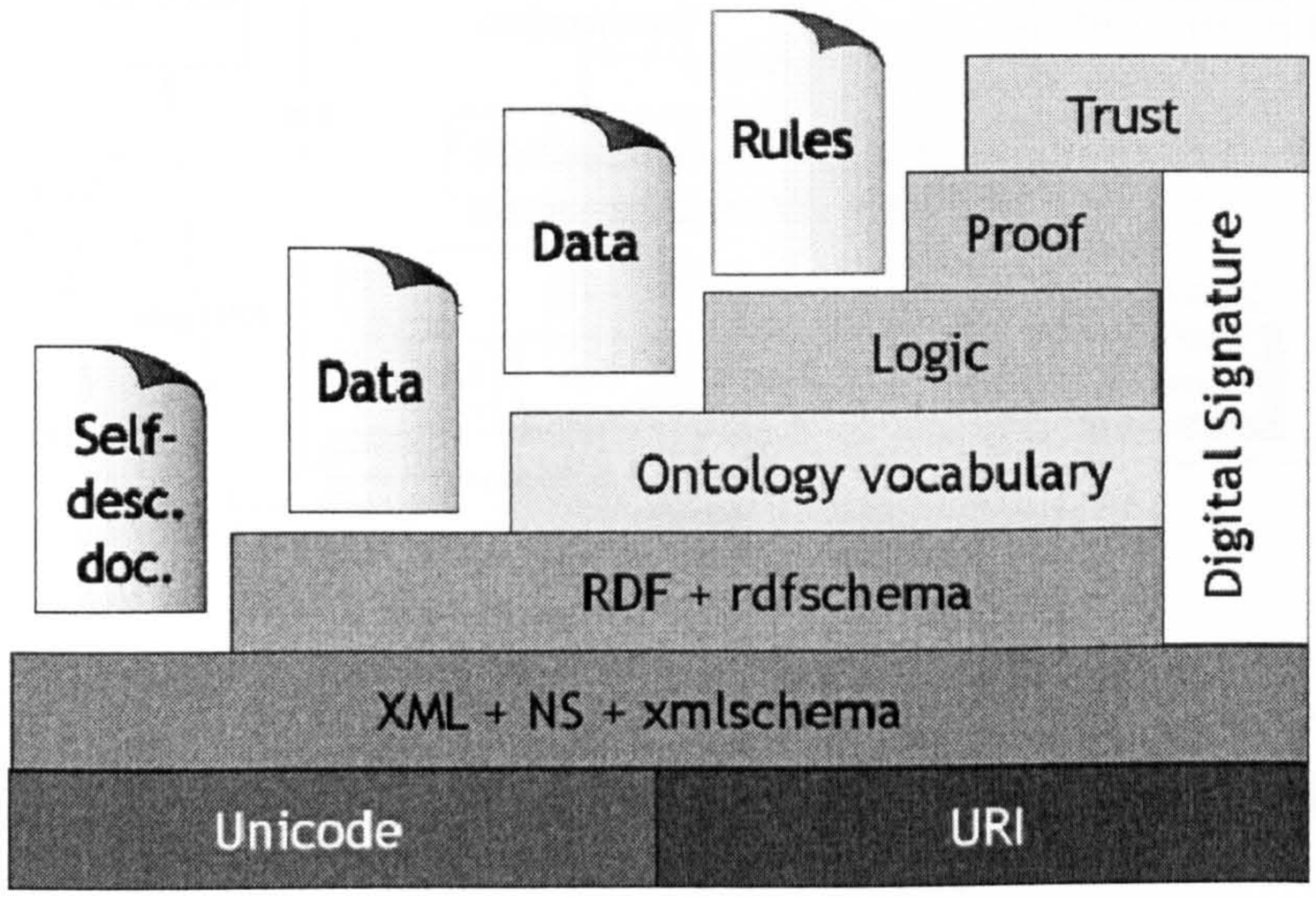


Figure 2-1: The Semantic Web Tower (Berners-Lee, 2002)

This architecture has been generally accepted and is expected to be developed over the next ten years. The involved techniques, including Uniform Resource Identifier (URI) and Unicode, Namespaces (NS), XML (Extensible Markup Language), XML Schema, Resource Description Framework (RDF), RDF Schema, Ontology and vocabulary, Logic, Proof, Trust and Digital Signature, are under fast development. Some layers of this 'tower' have standards (see Table 2-1) developed by the World Wide Web Consortium (W3C). Berners-Lee (1998) presented a brief plan (see Figure 2-2) to illustrate the interactions between major techniques.

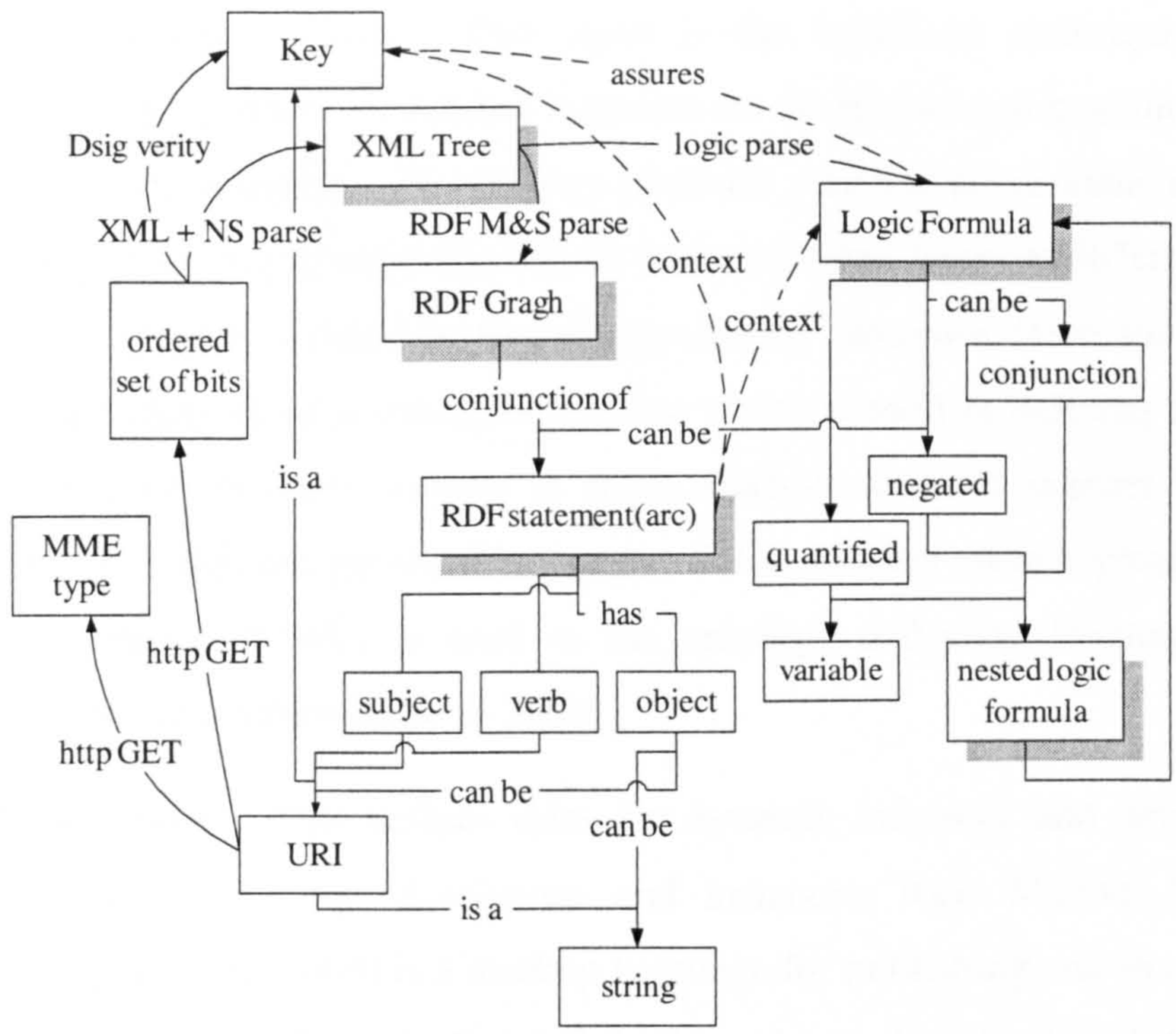


Figure 2-2: A Key Loop in the Semantic Web (Berners-Lee, 1998)

URI and Unicode Layer: They provide a means of identifying resources. Every data object and every data schema/model must have a unique and universal means of identification so as to be accessed by applications. These identifiers are called URIs (Uniform Resource Identifiers). Semantic languages, such as Extensible Markup Language (XML), Resource Description Framework Schema (RDF Schema), Web Ontology Language (OWL) and Semantic Web Rule Language (SWRL), all use URIs to point to things. URI and Unicode with NS (Namespaces) are already present in the current World Wide Web. URI became a W3C Recommendation in 1989;

XML + NS + XML Schema Layer: XML provides the common syntax, while Namespace (NS) and XML Schema define contents and rules. Details are provided in Section 2.2.2. XML became a W3C Recommendation in 1998;

RDF and RDF Schema Layer: RDF is a conceptual data layer on top of XML. It is application and domain neutral, and defines a metadata layer and domain-specific vocabulary. An RDF model can be used to describe anything that has a URI. Details are provided in Section 2.2.3. RDF became a W3C Recommendation in 1999;

Ontology Vocabulary Layer: This layer is the backbone technique for the Semantic Web. It provides a common language on the human and machine level to enable knowledge exchange. An ontology provides machine-processable semantics of data and information sources that can be communicated between different agents (Fensel, 2001), thereby facilitating knowledge sharing and reuse. Here, the ontology means the specification of a conceptualisation, which is used in defining terms and relationships between them, usually in some machine-readable manner (Hendler, 2001). Further details are provided in Section 2.2.4. In most current projects, Web Ontology Language (OWL) is used as the ontology definition language. OWL became a W3C Recommendation in 2004;

Logic Rules Layer: This defines rules for dynamic inference and definition of hierarchies and processing of schemas and instances. Rule Markup Language (RuleML) (Boley et al, 2005) is a markup language for publishing and sharing rules bases on the World Wide Web. RuleML builds a hierarchy of rule sub-languages upon XML, RDF, XSLT, and OWL. Semantic Web Rule Language (SWRL) was proposed in 2004 (Horrocks et al, 2004) based on a combination of some sub-languages of the OWL with some sub-languages of the RuleML. SWRL is under development now. In 2006, W3C published a recommended working draft of "RIF (Rule Interchange Format) Use Cases and Requirements", which is the first W3C recommendation for the logic layer. RIF specifies a format that enables rules to be translated between rule languages and thus transferred between rule systems;

Proof and Trust Layers: These involve the rating of sources and processes, and the monitoring of logical steps. Proof is the provision of explanation - why was certain knowledge inferred. Trust is an attribution of metadata statements, which make those assertions. For example, assertions about amazon.com are more trusted by a customer than those of an eBay seller. PML is a Proof Markup Language from the Knowledge Systems Laboratory at Stanford University (Pinheiro da Silva et al, 2006). The Trust layer will emerge through the use of a **digital signature.** Web users can add a digital signature to existing documents, which adds the logic of trust to a reasoning system. This means that the reasoning engine will have to be tied to the signature verification system. Documents will be parsed into two assertions trees: one is the content; the other is who has signed what assertions. Then, in the inference process, proof validation will check the signature on documents as well as

check the logic rules. A digital signature is coded in XML syntax. It became achievable when RDF was developed to support a proof language. In the W3C, research into digital signatures came from many directions, including the experience of DSig1.0 signed "pics" labels (W3C, 1999), and various submissions for digitally signed documents (W3C, 2006 a).

The proof and trust models and languages are the most embryonic. They will not take effect before the lower layers, on which they built, are clear enough. Therefore, in this research, these issues were left to future studies.

The major technology layers are introduced in Table 2-1, and described in greater detail below.

Table 2-1: W3C Recommendations for the Semantic Web

W3C Recommendations	First version	Latest version	Links
RIF (Rule Interchange Format)	2006	2006	http://www.w3.org/TR/rif-ucr
OWL (Web Ontology Language)	2004	2004	http://www.w3.org/2004/OWL
RDF (Resource Description Framework)	1999	2004	http://www.w3.org/RDF
XML- Signature	2001	2002	http://www.w3.org/Signature/
XML (Extensible Markup Language)	1998	2006	http://www.w3.org/XML
URI (Uniform Resource Identifier)	1989	2001	http://www.w3.org/Addressing

The Semantic Web changes the way information is presented online so as to enable the interoperability of diverse information sources. XML(S), RDF(S) and Ontology are the major techniques used to represent information.

2.2.2 XML and XML Schema

XML provides a standard mechanism to structure data (Ding et al, 2002), designed especially for Web documents. XML is a meta language which describes a way of defining a user's own set of tags and attributes (see Listing 2-1 for syntax).

```
<tag> attribute <tag>
```

Listing 2-1: XML Syntax

Therefore users can share the definition, transmit the data, validate and interpret them between applications and between organisations, although designers create their own customised tags separately. For example, a project manager defines a document entitled "project execution plan" (see Listing 2-2). It can be parsed by a contractor's software.

```
<?xml version="1.0"?>
<ProjectDocuments>
<item>
<title>    Project Execution Plan    </title>
</item>
</ProjectDocuments>
```

Listing 2-2: A Project Document in XML

2.2.3 RDF and RDF Schema

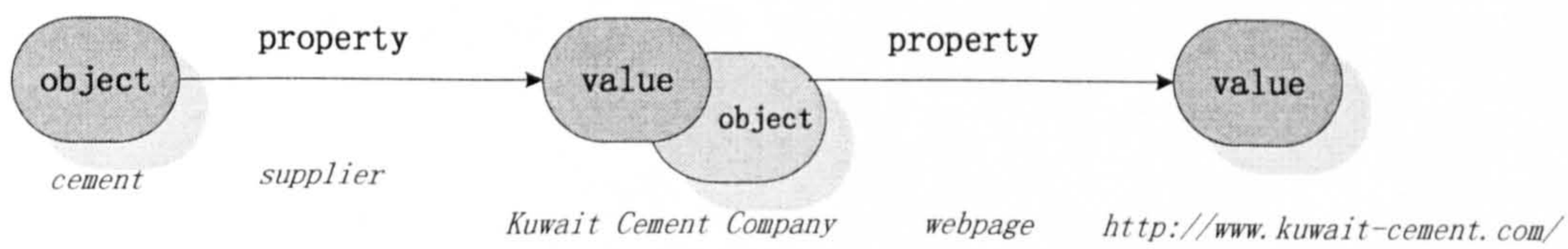
RDF is a framework for representing information on the Web. It provides a format, which states something about the data by making assertions (Hjelm, 2001). RDF describes resources in the model consisting of object, property, and value (as in Listing 2-3). Here resources are any objects (and some values) that can be uniquely identified by a URI. Properties express the values associated with the resources, which can be of different types. The names and the types of objects and properties are defined in a vocabulary. Everyone can design their own vocabularies following the XML and RDF syntaxes.

As shown in Listing 2-3, the statement that *the supplier of cement is the Kuwait Cement Company* with a specific Uniform Resource Locator (URL, a type of URI), can be represented in the RDF model as follows:

```
Cement, supplier, "Kuwait Cement Company",
Kuwait Cement Company, webpage, "http://www.kuwait-cement.com/"
```

Or be written as:

```
Supplier (cement, Kuwait Cement Company),
Webpage (Kuwait Cement Company, http://www.kuwait-cement.com/)
```



Listing 2-3: RDF Triple Model

The RDF data are coded in XML and woven together into a Web, as illustrated in Figure 2-3.

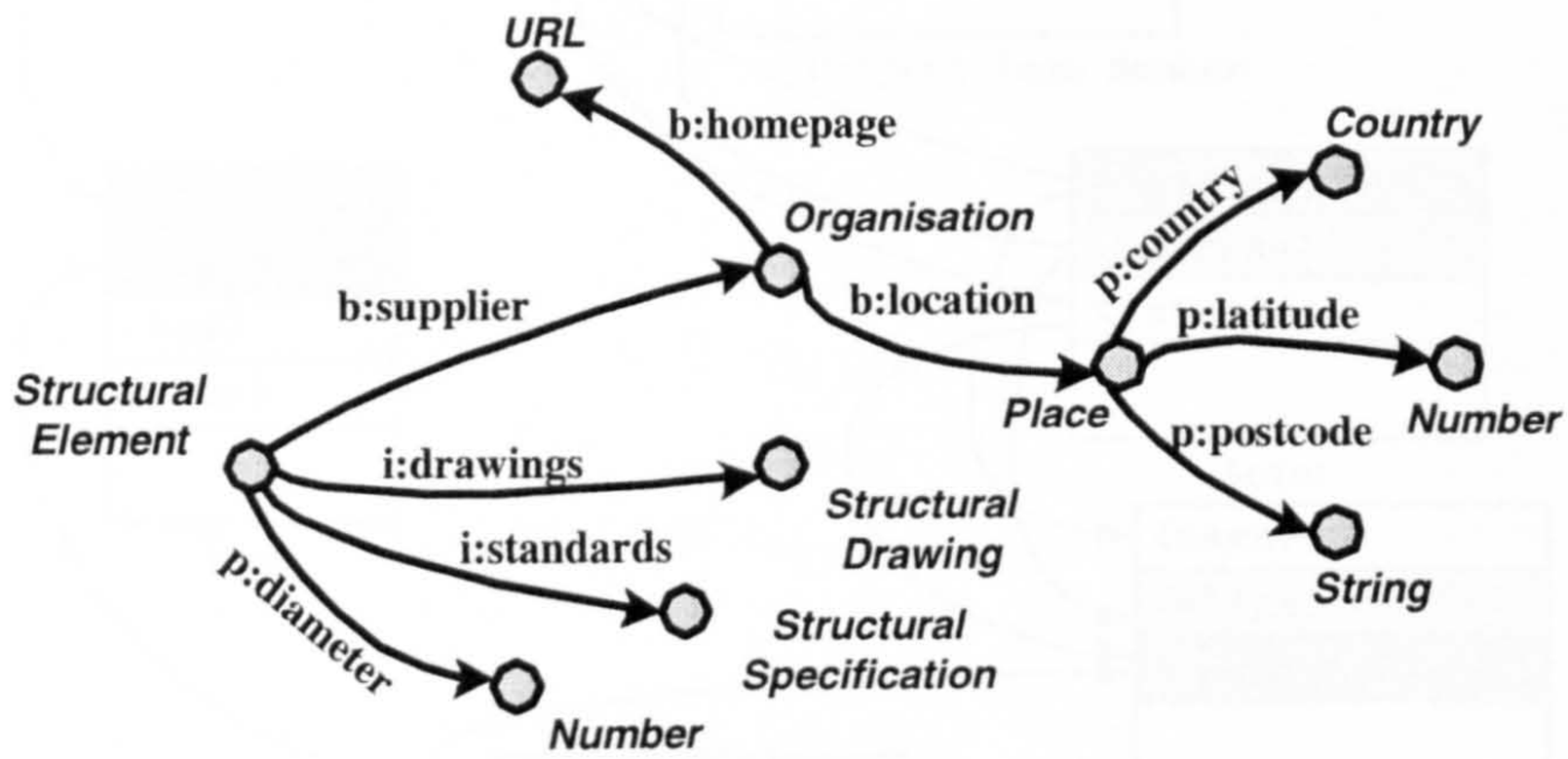


Figure 2-3: RDF Form

Every element uses links to join the Web (Berners-Lee, 2003). For instance, if five independently-developed databases, which are *Task*, *Project Team Member*, *Actor*, *Supplier* and *Structural Element*, are put on the Web (as in Figure 2-4), they are linked by semantic links which allow queries on one database to be converted into queries on another. The "who" in the *task* database means the same as "name" in a *Project Team Member* database. Similarly, "zip" in the *Project Team Member* database corresponds to "address" in the *Actor* database and "where" in the S*upplier* database. Other links are as shown by the arrows. Given this information, a search for "the partners involved in a certain task" can be retrieved from the *Task* database to the *Project Team Member* database and the *Supplier* database. All the links are represented by the RDF equivalent property.

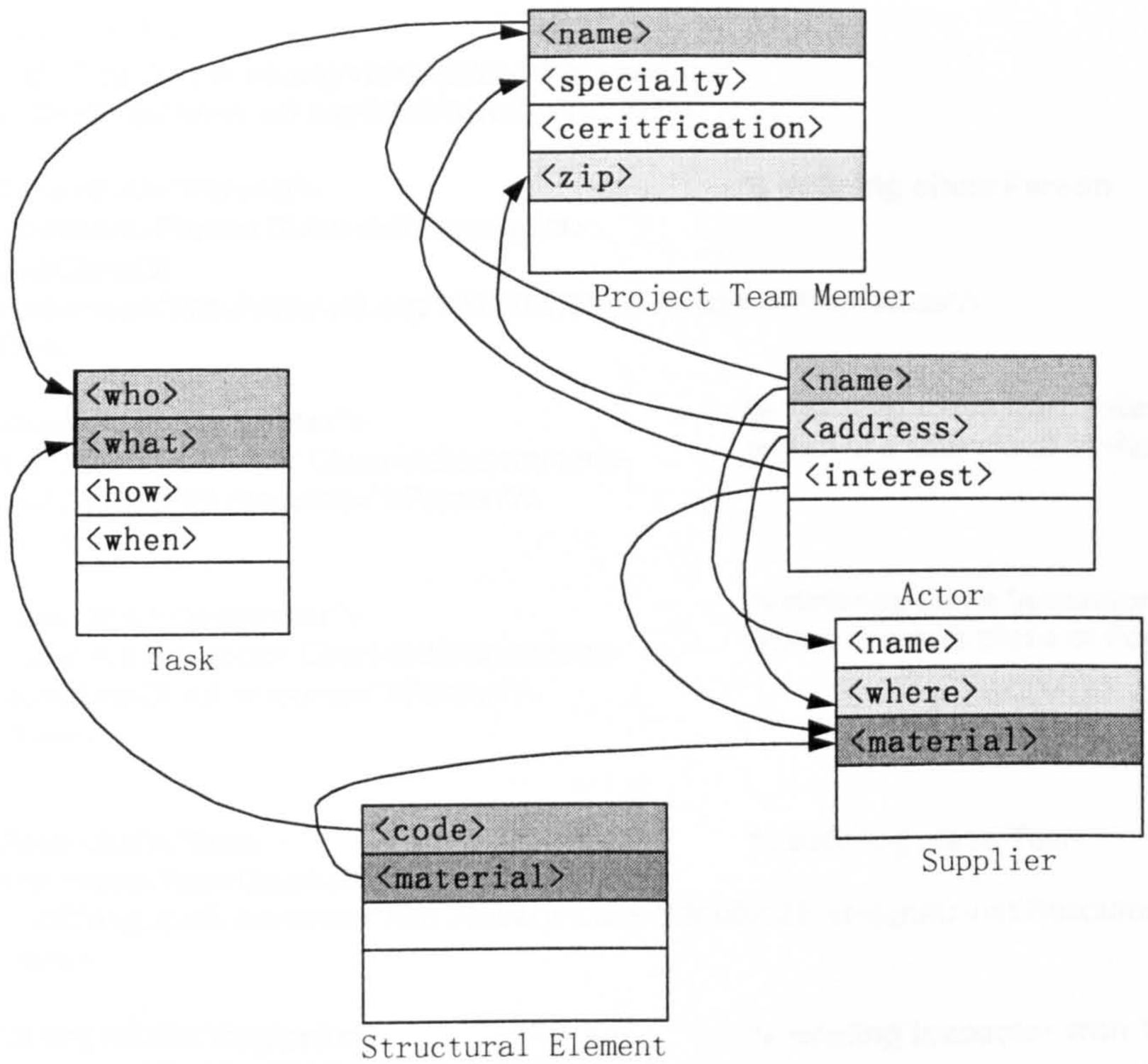


Figure 2-4: Semantic Links of RDF Data

RDF Schema (RDFS) provides a mechanism to define specific properties, in terms of the classes of resources to which those properties can be applied. The mechanism consists of class definitions and sub-class-of statements for class hierarchies, property definitions and sub-property-of statements for property hierarchies, domain and range statements for combinations and possible combinations of properties and classes, and type statements for resource of a specific class. Among them, the property is central to the definition, instead of the object and the class. Listings 2-4 to 2-5 show an RDFS instance about the carpenters group involved in the installation of a beam.

```
<?xml version="1.0"?>
  <xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">

  <rdfs:Class rdf:ID="Person">                                  % defining class Person
   <rdfs:comment>Person Class</rdfs:comment>
   <rdfs:subClassOf
       rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Resource"/>
  </rdfs:Class>

  <rdfs:Class rdf:ID="Carpenter">                               % defining class Carpenter,
   <rdfs:comment>Carpenter Class</rdfs:comment>                 which is a sub class of Person
   <rdfs:subClassOf rdf:resource="#Person"/>
  </rdfs:Class>

  <rdfs:Class rdf:ID="Inspector">                               % defining class Inspector,
   <rdfs:comment>Inspector Class</rdfs:comment>                 which is a sub class of Person
   <rdfs:subClassOf rdf:resource="#Person"/>
  </rdfs:Class>


  <rdfs:Class rdf:ID="Task">                                    % defining class Task
   <rdfs:comment>Task Class</rdfs:comment>
   <rdfs:subClassOf rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Resource"/>
  </rdfs:Class>

  <rdf:Property rdf:ID="Inspector">                             % relating Inspector with Task
    <rdfs:comment>Inspector of a Task</rdfs:comment>
    <rdfs:domain rdf:resource="#Task"/>
    <rdfs:range rdf:resource="#Inspector"/>
  </rdf:Property>

  <rdf:Property rdf:ID="Carpenter">                             % relating Carpenter with Task
    <rdfs:comment>List of Carpenters of a Task in alphabetical order</rdfs:comment>
    <rdfs:domain rdf:resource="#Task"/>
    <rdfs:range rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Seq"/>
  </rdf:Property>

  <rdf:Property rdf:ID="name">                                  % both Person and Task have name
    <rdfs:comment>Name of a Person or Task</rdfs:comment>
    <rdfs:domain rdf:resource="#Person"/>
    <rdfs:domain rdf:resource="#Task"/>
    <rdfs:range rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Literal"/>
  </rdf:Property>

  </rdf:RDF>
```

Listing 2-4: RDF Schema Example - Class/Concept Definitions

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"

 <Task rdf:ID="beam_A001">                          % name of the group numbers who
   <name>Programming XML in Java</name>             % set up beam_A001
   <inspector>
      <Inspector rdf:ID="ab">                       % Inspector ab's name is Amy Brown
        <name>Amy Brown</name>
        </Inspector>
   </inspector>
   <carpenter>                                      % Carpenter cd and ef's names are
      <rdf:Seq>                                     % Callio Derby and Elle Freeman
         <rdf:li>
           <Carpenter rdf:ID="cd">
               <name>Callio Derby</name>
             </Carpenter>
         </rdf:li>
         <rdf:li>
           <Carpenter rdf:ID="ef">
               <name>Elle Freeman</name>
             </Carpenter>
         </rdf:li>
      </rdf:Seq>
   </carpenter>
 </Task>
</rdf:RDF>
```

Listing 2-5: RDF Schema Example - Carpenter Group Members (Instances)

Both RDF and RDF Schema are based on XML and XML Schema. They provide a simple mechanism to represent knowledge for Web resources.

2.2.4 Ontology Vocabulary

The aim of the Semantic Web is to transform the current World Wide Web so that the information and services are accessible, processable and interpretable by computers as well as human beings. That will create an environment where software agents can readily perform sophisticated tasks and help humans find, understand, integrate, and use information. The key distinguishing feature of the Semantic Web will be the ontology. It is at the heart of the Semantic Web tower (Figure 2-1), and enables software agents to find the meaning of the information on Web pages by following hyperlinks to definitions of key terms and rules for reasoning about them logically.

A Number of possible languages can be used to program ontologies, including Prolog (Programming in logic), OKBC (Open Knowledge Base Connectivity)

model, KIF (Knowledge Interchange Format), CL (Common Logic), Loom, DAML+OIL (DARPA Agent Markup Language and Ontology Inference Layer) and OWL (Web Ontology Language). These language specifications were developed over several years. Among them, DAML+OIL and OWL are the two major ones, with OWL the most widely adopted.

OWL (W3C, 2004 c) is a practical ontology language, which is being adopted in many Semantic Web applications. The W3C recently recommended the OWL language and RDF for building Web ontologies. As such, OWL is rapidly replacing its predecessor DAML+OIL (DARPA, 2006). OWL was adopted in this research as a language to describe the class of information concepts/properties and relations between classes/properties that are inherent in construction project documents and applications. Listing 2-6 is a simple instance describing a building using OWL.

```
class-def building                      % building is a class
class-def people                        % people is a class
    subclass-of NOT building            % of things that are not building
class-def place                         % place is a class
    subclass-of NOT building            % of things that are not building

class-def architect                     % architect is a class
    subclass-of people                  % which is a subclass of people
    slot-constraint design
        value-type building             % that design building

class-def business center               % business center is a class
    subclass-of place                   % which is a subclass of place
    slot-constraint occupied
        value-type business people      % that locate in business center

class-def business tower                % business tower is a class
    subclass-of building                % which is a subclass of building
    slot-constraint locate
        value-type business center      % that locate in business center
```

Listing 2-6: Ontology for Building

A specific ontology stands for knowledge in a specified domain. Generally, an ontology provides a shared and common understanding of a domain that can be communicated between people and application systems. They are heterogeneous and distributed information sources.

Theoretically, everyone can develop their own ontology. This leads to the issue of ontology management. Ontology management includes ontology creation, reuse, merger, storage, retrieval and maintenance. Lee and Goodwin (2006), who work in the IBM T. J. Watson Research Center in Hawthorne, are developing an Ontology Management System for electronic markets. For the combined use of ontology in collaborative working, many techniques are important: combining, merging/integrating, aligning, mapping articulation, translating and transforming. The key idea of ontology management techniques is matching *equivalent* concepts between ontologies (Hitzler et al, 2005). Researchers also put efforts into mechanisms for avoiding mismatches (de Bruijn et al, 2006). In this research, the basic ontology management issue was addressed, and ontology merger was taken as an example of ontology management techniques. Please find details in Chapters 6 and 7.

2.2.5 Summary

The Semantic Web is an extension of the current World Wide Web. Parts of it, the URI, XML, and RDF, have been developed and are used to build the current Web. Based on these, RDF Schema and Ontologies present additional attributes for the Semantic Web. Further attributes, such as the reasoning inference, proof and trust, will be addressed with the development of the Semantic Web. Generally, there are five basic steps to use the Semantic Web:

(1) Using URIs as a global naming scheme;

(2) Using XML and RDF as a standard syntax for describing data;

(3) Using RDF Schema as a standard means of describing the properties of that data;

(4) Using Ontology as a standard means of describing relationships between data items;

(5) Using Rule markup languages and rule-based systems as the means to undertake logic reasoning and support trust and security.

It was predicted that "science, engineering and commerce will benefit enormously from the Semantic Web" (Berners-Lee, 2003). However, research, standardisation and field work are still needed to ensure it happens. Current applications mainly focus on knowledge management and Web Services. Meanwhile some prototype

systems are being developed for industrial deployment; these are reviewed in the next section.

2.3 SEMANTIC WEB APPLICATIONS

In this section, the applications of the Semantic Web are discussed from three perspectives: industrial applications, potential applications in the construction sector and the key features, enablers, barriers, and potential benefits from adopting the Semantic Web.

2.3.1 Industrial Applications

The Semantic Web consists of layers of techniques (as shown in Figure 2-1) and the development of each layer is associated with the respective computing languages and software tools. These computing languages and tools are usually proposed and used in specific industrial projects. For example, Protégé is an ontology editor which was initially developed by Stanford Medical Informatics (2006) for biomedical ontologies and is now widely used in diverse academic/industrial projects. Although some projects stated they were using a semantic technique, for example the Ontology, they also involved other techniques, such as XML and RDF, because these techniques are not isolated or independent.

Wherever heterogeneous information exists, the Semantic Web could be applied to manage the information in that domain. It is hard to enumerate all the existing Semantic Web applications because it is one of the most popular subjects in recent years with so many projects being carried out. These projects cover diverse disciplines such as bio-informatics (Stephens et al, 2006), e-Commerce (Fensel 2001 and Benatallah et al, 2006), education (Henry et al, 2003), human science (Hyvönen et al, 2004), healthcare (Dogac et al, 2006) and engineering (Brown et al, 2004) to name a few. In this thesis, representative applications are reviewed.

Since the Semantic Web is a Web with meaningful information (Berners-Lee, 2001), and the term, knowledge, is used to describe "the application of data and information" (Ackoff, 1989), in this thesis Semantic Web applications are considered from two perspectives: Knowledge Management (KM) and information-intensive Web Services (WS). Most projects include both KM and WS but emphasise one aspect.

(1) Knowledge Management

Firstly, it is necessary to specify the definition of data, information and knowledge.

As defined by Ackoff (1989):

- **Data** are symbols. They are raw, existing in any form and do not have meaning of themselves.
- **Information** is data that have been given meaning by way of relational connection. In computing science, a relational database generates information from the data stored within it.
- **Knowledge** is information with relations and patterns. It is an appropriate collection of information. In computing science, most of the applications currently used (modelling, simulation, etc.), exercise some type of stored knowledge.

Based on the review by Hlupic et al (2002) of 18 different definitions of knowledge, Bouthillier and Shearer (2002) distinguished knowledge from information by stating that "knowledge is predictive and can be used to guide action while information merely is data in context". Then, Knowledge Management (KM) can be regarded as "the process of collecting, organising, classifying and disseminating information throughout an organisation, so as to make it purposeful to those who need it" (Albert, 1998). Knowledge Management can also be defined as the action to manage knowledge, such as "combining indexing, searching, and pushing technology to help companies to organise data stored in multiple sources and deliver only relevant information to users" (Hibbard, 1997). Some other definitions emphasise knowledge resources and assets (Quintas, 2005). For example, Anthes (1991) defined KM as "policies, procedures and technologies employed for operating a continuously updated linked pair of networked databases". Knowledge Management can also be regarded as a methodology "of creating a thriving work and learning environment that fosters the continuous creation, aggregation, use and re-use of both organisational and personal knowledge in the pursuit of new business value" (Cross, 1998). In this thesis, the Hibbard's definition is adopted. It can be seen that the more precisely information is provided to people, the better knowledge is managed and the more conveniently it is used.

Different from traditional knowledge management approaches, which process information on a document/text level and use an indexing system with a keywords-based search engine, the implementation of Semantic Web techniques, RDF(S), Ontology and Rules specifically, provides information-centric document management and an ontology-driven search engine by integrating ontological annotation with data (such as those in a document or database). These annotations define the meaning of information using ontological concepts instead of isolated keywords. A number of academic and industrial projects were associated with the technical layers in Figure 2-1, focusing on processing information syntactically, semantically, and inferentially. They provided services such as Web data mining (Norguet et al, 2006), information retrieval (Kiryakov et al, 2004), knowledge sharing (Davies et al, 2003), intelligent notification (Kwon et al, 2005), collaborative working (Lewis et al, 2004), and decision support (Kwon and Kim, 2004). The following selected projects demonstrate some typical Semantic Web applications from the knowledge management perspective:

Ontobroker (Ontoprise, 2006) was one of the most famous projects in the early days of the Semantic Web. It has been commercialised by the Ontoprise Company. Ontobroker was an inference engine, which could be used to reason about instances of ontology or ontology schema. Reasoning about instances of ontology involved deriving a certain value for an attribute applied to an object. These inference services provided stronger support to databases than SQL query engines. To intelligently access knowledge, the Ontobroker project used ontologies to annotate and wrap Web documents, and provided an ontology-based answering service. It supported clients who sought knowledge and who wanted to enhance the accessibility of their Web documents. Compared to traditional information retrieval, Ontobroker realised knowledge-based reasoning on the Web. For example, ProPer is a human skills ontology, which was used to manage human resources on Ontobroker. People can be found by searching for a competence, such as skills, working experiences, and language abilities through their own documents, which were annotated according to ProPer ontology. These documents might be homepages, project reports, and CVs of people inside or outside a company. This is an efficient mechanism to manage the skills/competencies of people for a company.

MIAKT (Medical Imaging and Advanced Knowledge Technologies) (OpenClinical, 2004) was a collaborative project between the University of Oxford, Southampton, Sheffield, King's College London, Open University and Epistemics Ltd., which focused on collaborative diagnosis and plan of action between multi-disciplinary medical expertise to resolve problems in the domain of breast cancer. The project proposed a breast cancer imaging ontology and applied it in medical images and signals management. A series of software were developed in this project which can provide services of image and other multimedia analysis, automatic generation of reports from ontological instances, search and retrieval of breast-cancer related instance information, image registration of those taken at different times, classification of regions of interest that are marked up on images, and analysis of regions of interest.

OntoWeb (2003) was a European Union founded project that involved a community of researchers and industry in the study of Ontology-based information exchange for knowledge management and electronic commerce. It set up a thematic network, which provided a standard portal allowing registered members to upload documents, images and files to the site. Then they could join Special Interest Groups (SIGs), share resources and attend workshops. Five SIGs were available:

SIG1: Ontology-based content standardisation and content harmonisation;

SIG2: Ontology language standards;

SIG3: Enterprise-standard ontology environments;

SIG4: Industrial applications;

SIG5: Language technology in ontology development and use.

OntoWeb was not only a platform for communication, but also proposed an ontology to manage the shared knowledge. The OntoWeb ontology included concepts of deliverable, event, milestone, news, organisation, person, product, project publication, and work package. These concepts and their sub-concepts annotated knowledge from diverse resources, such as the OntoWeb events or documents submitted by members, so as to share information and knowledge within the community. Figure 2-5 displays the interface of the OntoWeb knowledge search

engine. It searches for the semantics of information, such as the properties of deliverables in this figure.

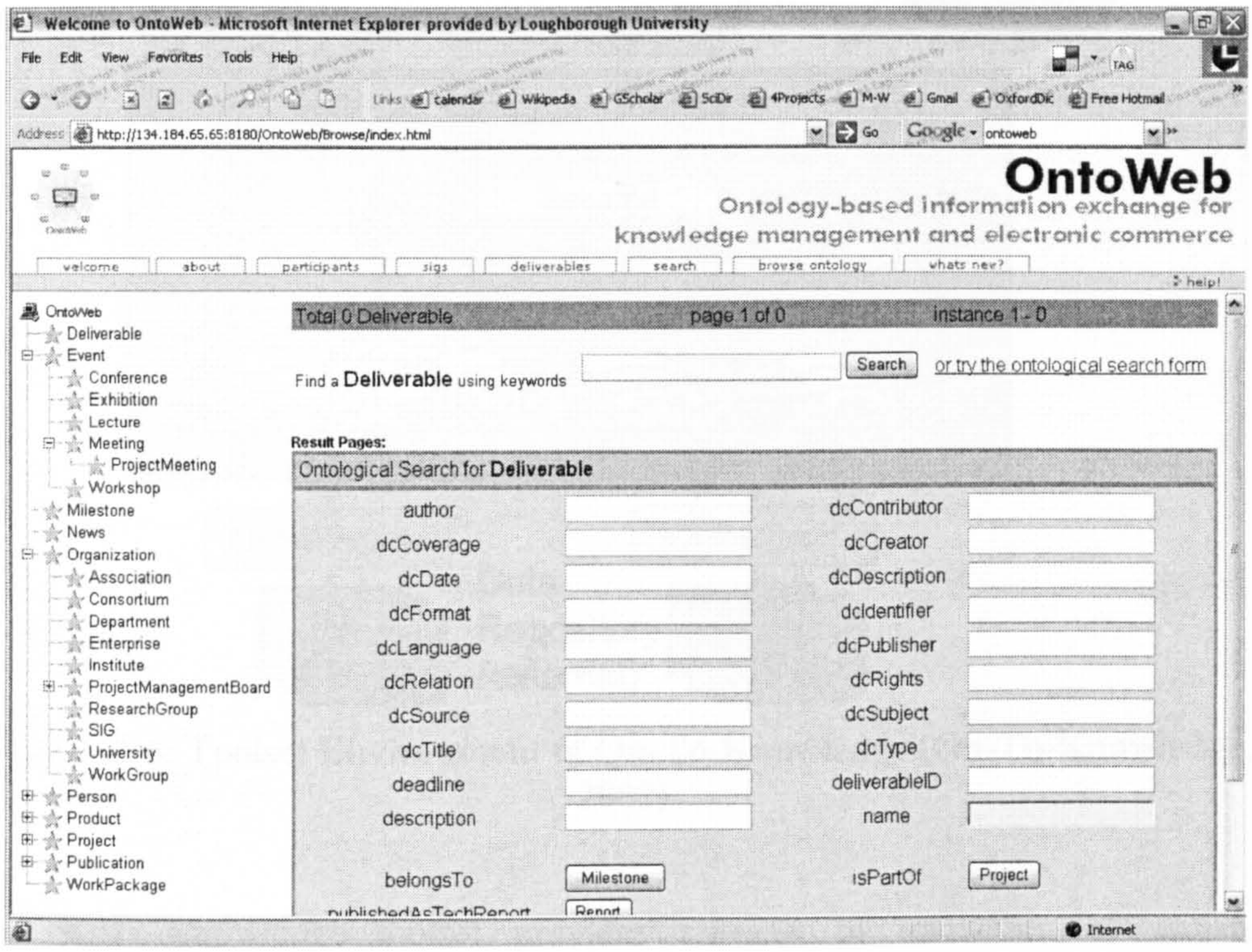


Figure 2-5: OntoWeb Search Interface (OntoWeb, 2003)

On-To-Knowledge (2002) was a project in the Information Society Technologies (IST) Program, which built an ontology-based tool environment to speed up knowledge management, dealing with the large numbers of heterogeneous, distributed, and semi-structured documents typically found in large company intranets and the World Wide Web. The major outcomes of On-To-Knowledge were a methodology which presented knowledge efficiently and effectively, an intelligent search tool, and three case studies. The On-To-Knowledge project partners developed a toolset including OntoExtract, OntoWrapper, Sesame, OntoShare, OMM, BOR, OntoEdit, Spectacle and QuizRDF. The integration of these tools (Figure 2-6) enabled different participants (organisations, individuals, or departments) to maintain their own knowledge structure while exchanging information.

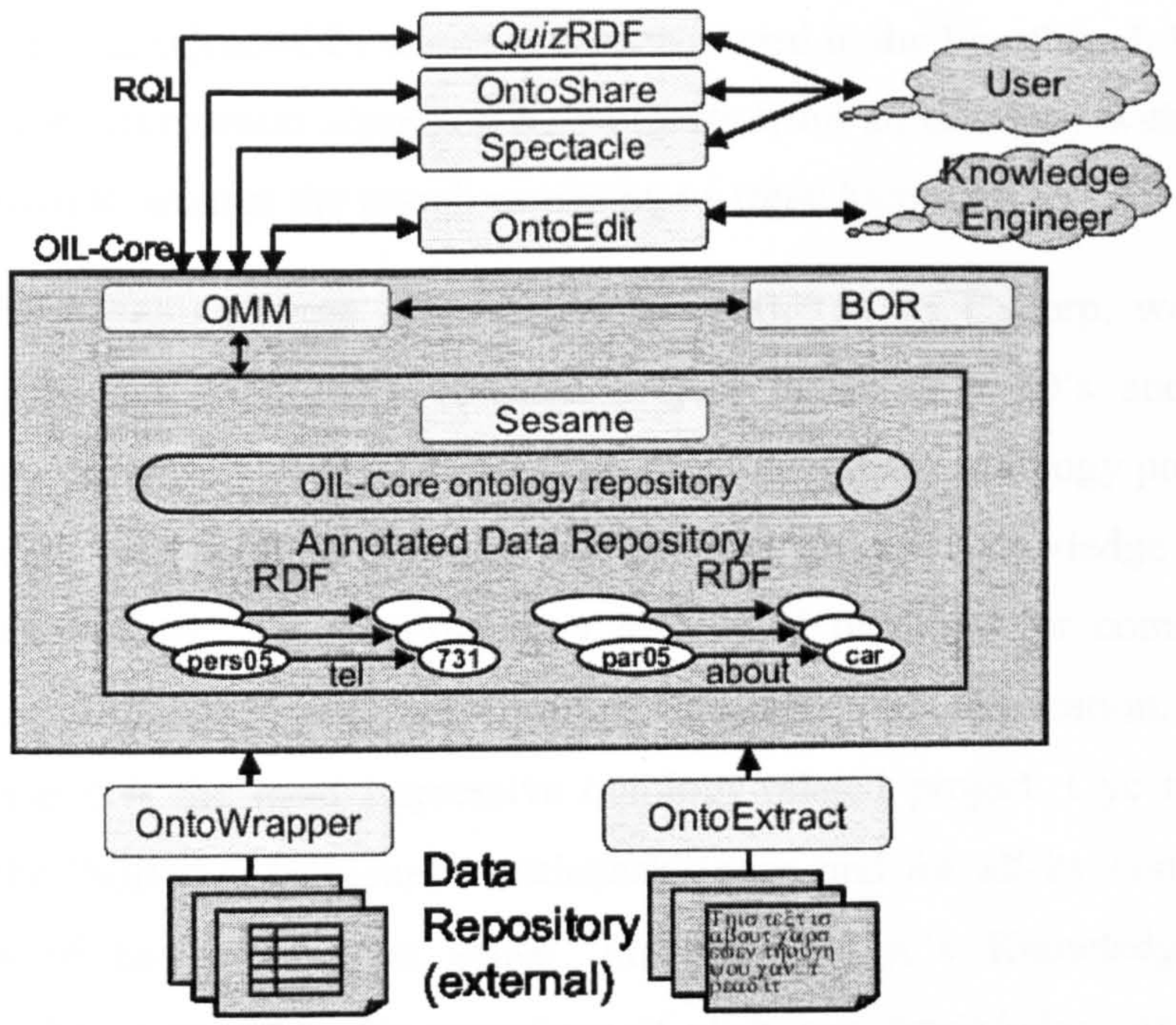


Figure 2-6: Toolset Environment of On-To-Knowledge (On-To-Knowledge, 2002)

The On-To-Knowledge toolset provides services of extraction of semantic information from textual information (OntoExtract and OntoWrapper), ontology editing (OntoEdit), ontology storage and retrieval (Sesame), versioning, tracking changes, fine-grained access control, meta-information, and multiple integration options (OMM, Ontology Middleware Module), DAML+OIL reasoner (BOR), ontology-based information navigation and querying (QuizRDF), and ontology-based visualisation of information (Spectacle). The tools were applied in three case studies:

- Ontoshare was used by British Telecom Call Centre to share best practice information between relevant call centre agents;
- OntoExtract was used in ontology extraction from documents, such as CVs, recruitment profiles, course, project descriptions, and IAS (International Accounting Standards) documents on the global Swiss Lift Intranet so as to retrieve relevant information, reformulate insufficient retrieval results, and compare information with associated concepts from the ontology, such as skills descriptions among employees;

- The $RDF_{Ferrent}$ (QuizRDF) search engine was used in the EnerSearch Web site to search for information annotated by concepts from an ontology with Spectacle being used to arrange the search results into a topic hierarchy.

Cyc (Cycorp, 2006) (comes from en-cyc-lopedia), run by Cycorp, was initiated from an Artificial Intelligence research project in the early 80's and has now evolved into one of the most influential information systems ontology projects. The Cyc project focuses on the presentation of, and reasoning on, knowledge rather than applications, and the Cyc ontology is a high level ontology for common sense knowledge. Although Cyc is not a typical Semantic Web application, it is listed here because it is the most impressive ontology-related project. Cyc is aimed at breaking the "software brittleness bottleneck" once and for all by constructing a foundation of basic "common sense" knowledge. Cyc's Knowledge Base is intended to be an encyclopedic repository of all human knowledge. As such, it is devoted to using ontologies to represent facts and the inscription of rules about all existing and imaginable things. Cyc has proposed six ontologies to the DAML program Ontology Library, including the Cyc Knowledge Base Upper Ontology defining schema, the OpenCyc Knowledge Base Upper Ontology, North American Industrial Classification System codes, Cyc's Taxonomy of Transportation Devices, and OpenCyc Ontology in DAML and OWL. OpenCyc is the open source version of the Cyc(r) technology. It can be used as the basis for a wide variety of intelligent applications (refer to Section 2.4.2-(4)). Besides the ontology development, Cycorp also targeted at the ontology mapping issue. All of these are meaningful efforts for the Semantic Web.

The above Semantic Web applications prove that the Web could become a vast knowledge repository rather than a document collection. As such, Web would be able to provide more interactive and customised services to users. Some representative Web Services which are provided by the Semantic Web are briefly reviewed in the next section.

(2) Web Services

The traditional World Wide Web provided services such as information distribution, sharing and retrieval. The convenience of information access on the Web and the ease of information uploading to the Web led to the volume of online information

increasing tremendously. This resulted in the problem of information overload, loss and misunderstanding when Web users distribute, share and retrieve specific information because of the lack of links between related information over the World Wide Web. The appropriate and clear definition of information on the Semantic Web enables Web Services to meet user's specific requests in a timely manner. The W3C (2006 b) defined a **Web Service** as "a software system designed to support interoperable machine-to-machine interaction over a network". Web services allow people and organisations to communicate information without an intimate knowledge of the IT systems of its sources. Three selected Web Services projects are introduced:

Antoniou and van Harmelen (2004) introduced a **Semantic Web-based Elsevier journal paper indexing system.** This enabled information from different sources to be indexed and queried by using thesauri (a type of ontology for multi-disciplinary information) instead of a previous indexing system which divided journals into disciplines with each journal having its own indexing system, which led to difficulties in accessing interdisciplinary information. Experimental thesauri were being developed in the area of medical information and life sciences.

The **MuseumFinland** project (Hyvönen, 2004) provided a semantic portal to access the collection databases of several Finnish museums using the Semantic Web technology. A search engine was developed to query those distributed heterogeneous cultural information from the museum collections (e.g. artifact type, material, place of usage, time of creation, and situation of usage) using several cultural ontologies.

KMI PlanetOnto was a Web-based news server, which facilitated communication within a laboratory and allowed the wider community to access lab-related items of interest (Domingue and Motta, 2000). It provided personalised news feeds and alerts and proactively identified potentially interesting news items. Figure 2-7 is the architecture of PlanetOnto. It supported seven main activities: (1) story submission, (2) story reading, (3) story annotation, (4) provision of customised alerts, (5) ontology editing, (6) story soliciting, and (7) story retrieval and query answering. PlanetOnto was supported by an integrated suite of tools (such as KM Planet, Knote, Lois, Newsboy, Newshound, OCML and WebOnto, shown in the centre of Figure

2-7), which enabled ontology-driven document formalisation and augmentation, standard browsing and search facilities with deductive knowledge retrieval.

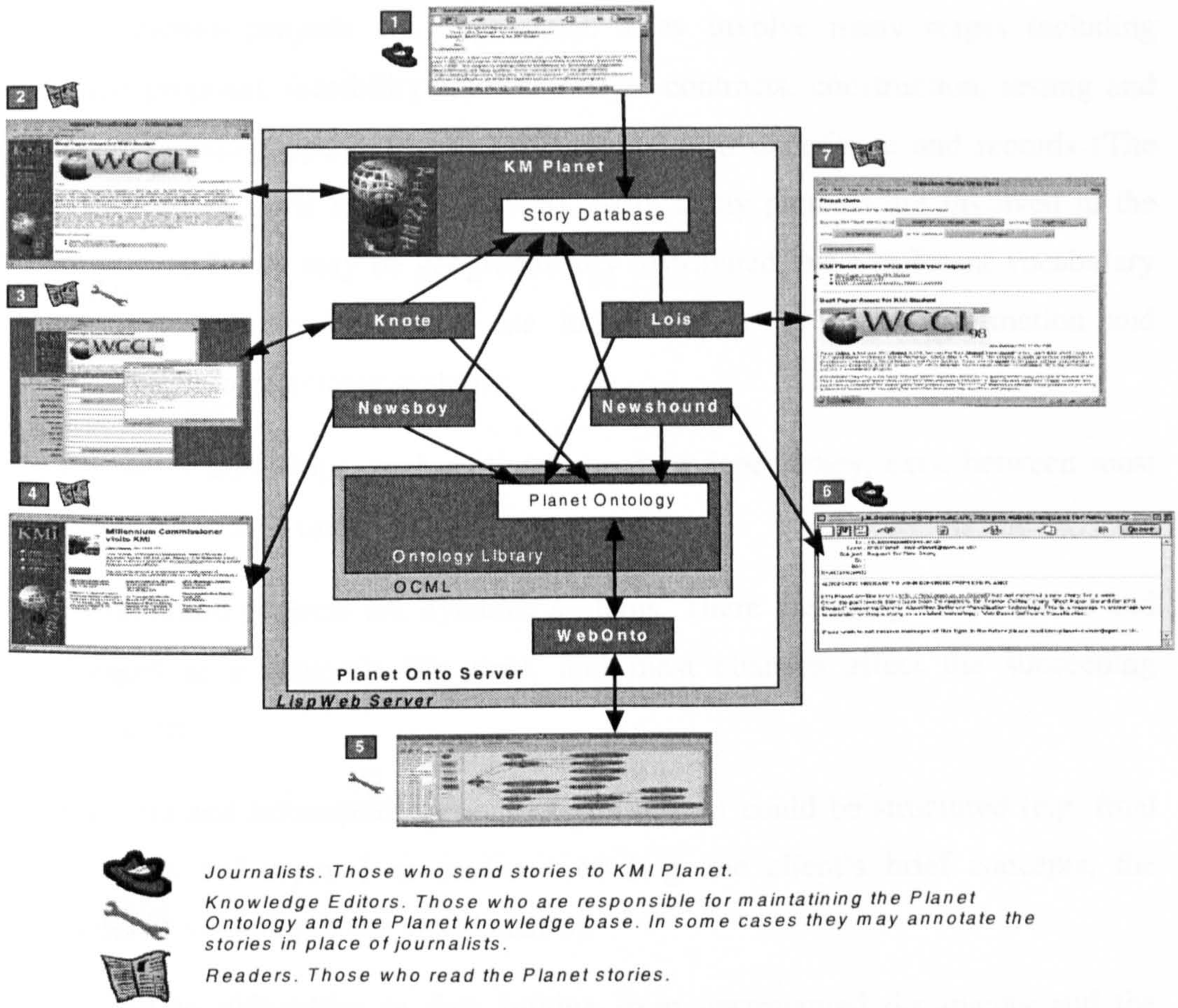


Figure 2-7: The Architecture of PlanetOnto (Domingue and Motta, 2000)

A common advantage of the above Semantic Web Services over traditional Web Services lies in its customised service based on ontological information.

2.3.2 Potential Applications in Construction Industry

Information management is becoming increasingly important in modern construction projects. Using Information Technologies (IT) to support project management and the company organisation is now well established. Web-based information management is an appropriate collaboration mechanism for the fragmented nature of the construction industry (Chan and Leung, 2004) and many Web-based systems are now available. However, Web-based electronic information management is a painstaking effort on a construction project for five reasons:

- Construction projects are creative and unrepeatable. Most project documents are not reusable;
- Construction projects are fragmented. They involve many stages including initial proposal, feasibility studies, design, contracts, construction, testing and commissioning, operation and maintenance, and experience and records (The Institution of Civil Engineers, 1996). Also, many partners are involved in the project, and they may be geographically distributed, have different vocabulary for project information, and use independently-developed information and knowledge management systems;
- Various relationships, such as time, space or dependency, exist between most construction activities;
- Construction projects are dynamic systems. There are many uncertainties and changes in a project's life time, and most changes affect the succeeding activities;
- The data and information generated in a project could be structured (e.g. final drawings and reports) or unstructured (e.g. the client's brief concepts, the architect's briefing notes and sketches).

These lead to difficulties in data mining from unstructured documents and the sharing of real time information between distributed project partners. Chapter 3 provides a detailed review of construction information management.

Generally speaking, as in other industries, the Semantic Web is good for providing Knowledge Management and Web Services at various levels and for various activities throughout a project's life cycle.

Some preliminary researches about Semantic Web techniques in the construction domain are ongoing. Researchers have used XML to establish a meta-data-based collaboration system model in order to substitute traditional Web-based collaboration in construction project management (Leung et al, 2003). Several XML-based vocabularies for Building and Construction were developed, such as bcXML (Building Construction Extensible Mark-up Language) (Frits et al, 2001), aecXML (Architectural, Engineering and Construction XML) (IAI, 2002) and

ifcXML (Industry Foundation Classes XML) (IAI, 2006 a). Recent studies that address the ontology issue include:

- The e-COGNOS (Methodology, tools and architectures for electronic COnsistent knowledGe maNagement across prOjects and between enterpriSes in the construction domain) project proposed a generic ontology for interoperation between the knowledge bases of construction enterprises (Lima et al, 2003);
- van Rees et al (2004) explored the practical use of the Semantic Web from an ontology perspective;
- van Rees and Tolman (2004) applied Semantic Web technologies to annotate building specifications and drawings;
- Cai and Ng (2004) intended to build an ontology for changes in projects;
- Lai and Carlsen (2004) developed a framework for ontology-driven collaborative design;
- As an extension of the e-COGNOS ontology, El-Diraby and Gill (2006) proposed a taxonomy for construction terms used in exchanging project risk information between construction and financial institutions;
- Ferreira da Silva et al (2006) discussed the interoperability (mapping concepts) between Semantic Resources; and
- El-Gohary and El-Diraby (2006) proposed a semi-automatic approach which used ontology merging techniques for supporting stakeholders' collaboration.

Of these projects, the **e-COGNOS** project is the most notable. It was an IST (Information Society Technologies) project, running from 2001 to 2003. This project aimed at specifying and developing an open model-based infrastructure and a set of tools that promote consistent knowledge management within collaborative construction environments. The outcome of e-COGNOS project consists of 3 products:

(1) An Ontology in DAML+OIL format.

(2) An Ontology Server (e-COSer) excluding the ontology browser.

(3) A Knowledge Management Infrastructure (e-CKMI), which integrated the ontology server with knowledge bases;

The e-COGNOS project adopted an ontology to integrate knowledge bases, and the e-COGNOS ontology committed to the IFC model (details of IFC model are provided in Sections 3.6.2-(2) and 3.6.3-(3)).

However, e-COGNOS intended to resolve the information inconsistency problem in sharing knowledge bases between companies, so it did not take advantage of other Semantic Web techniques, such as the Rule, Proof and Trust. Also, the search for documents on e-CKMI is less semantics-oriented because it is not concerned with a document's properties other than its ontological concepts in the search criteria. In an e-COGNOS search, concepts and index terms were used instead of keywords for document submission and search (as displayed in Figures 2-8 and 2-9). As the algorithm for calculating concepts/index terms for a document is mathematical (refer to Section 5.4.1), it requires a specialist, who has deep understanding of both ontologies and construction knowledge to build and maintain the ontology. Concepts and index terms provided more semantic information on documents to the e-COGNOS search engine, but not enough to specify diverse properties and relationships of documents and not explicit to end-users.

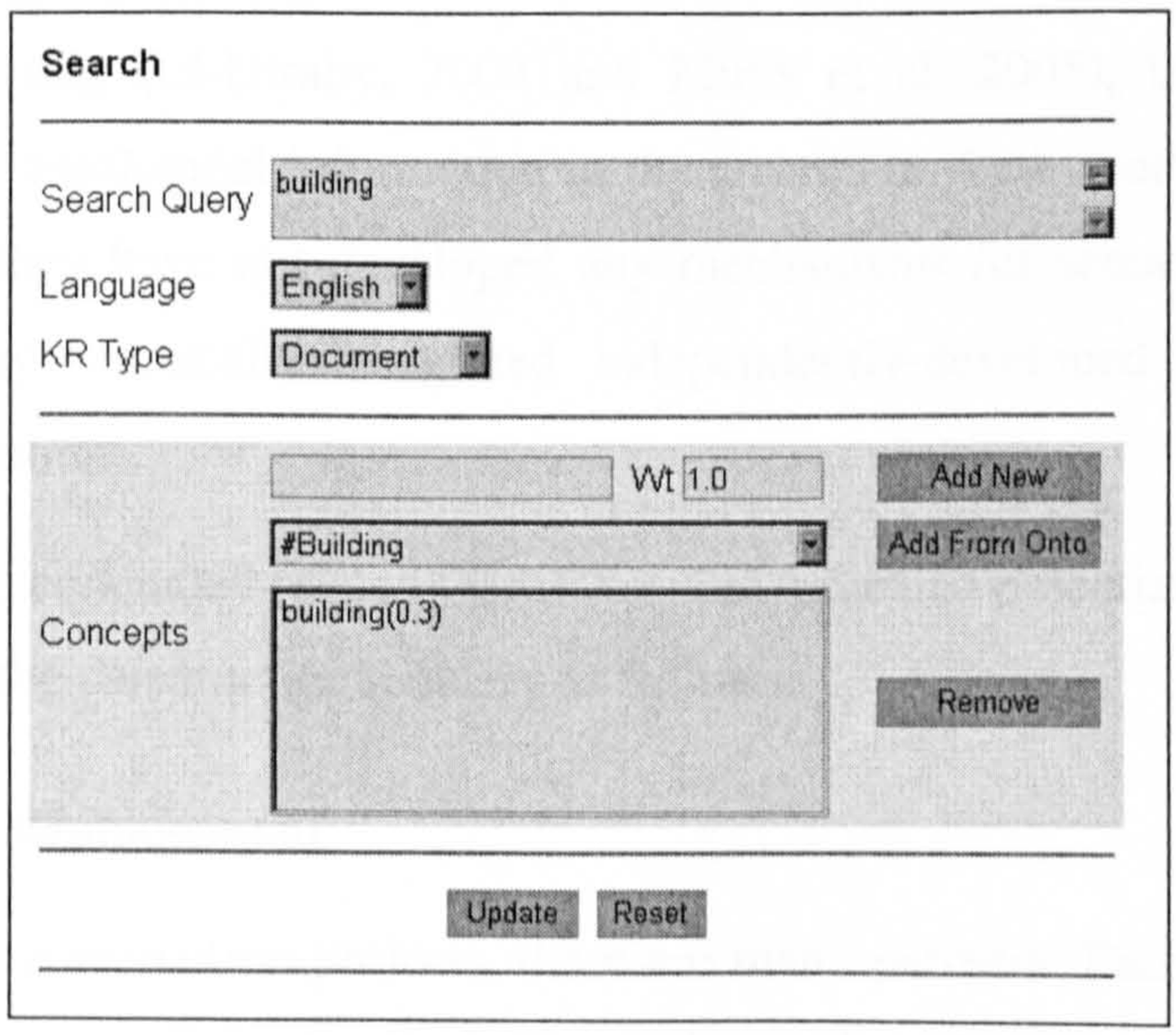


Figure 2-8: Interface of Search on e-COGNOS Server (e-COGNOS Consortium, 2002)

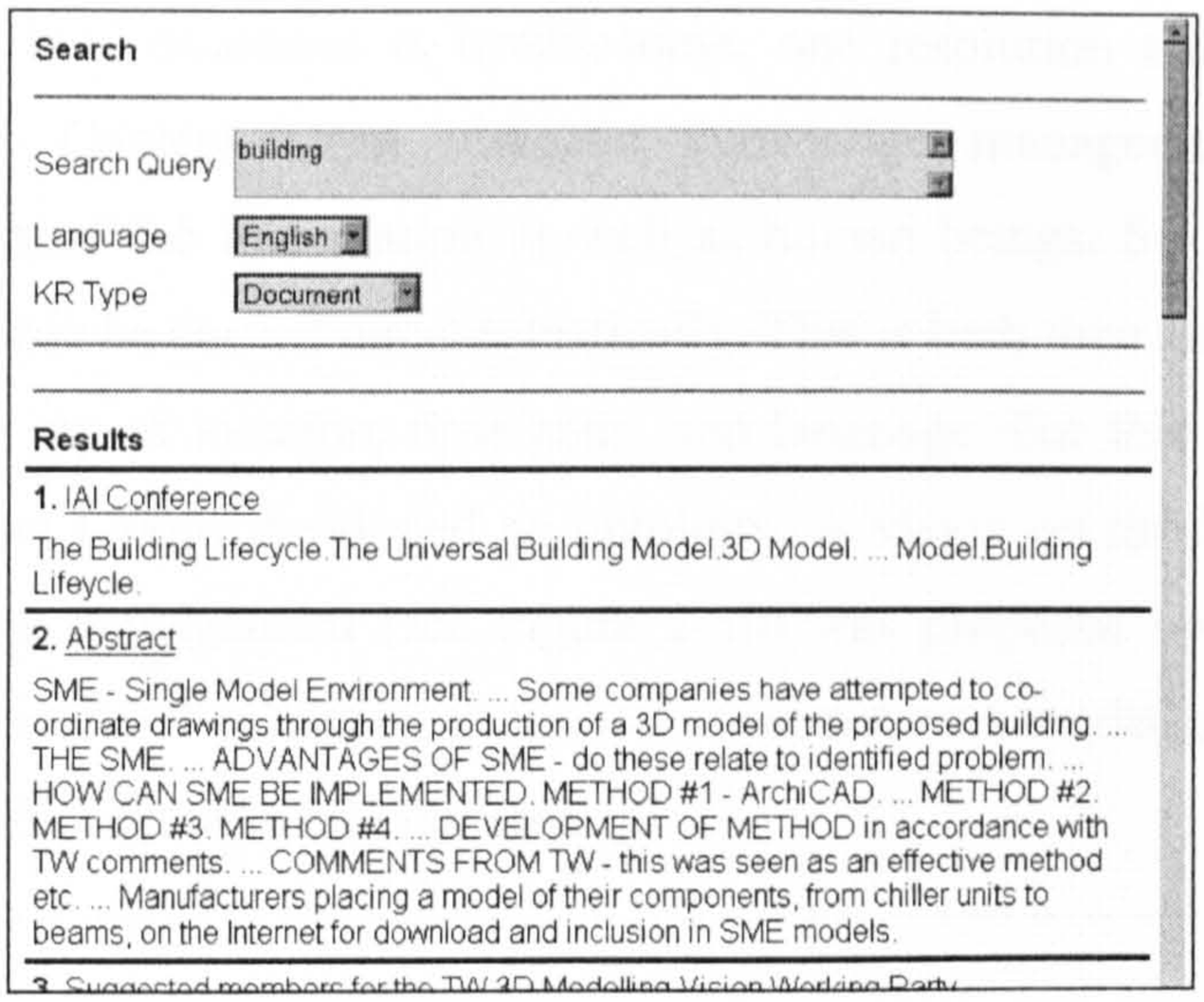


Figure 2-9: Interface of Search Result on e-COGNOS Server (e-COGNOS Consortium, 2002)

Because the Semantic Web is in its embryonic stage, most related research projects in the construction industry are high level frameworks for specific domains (Lai et al, 2003; Danso-Amoako et al, 2006) or fragmented ontologies (Lima et al, 2003; El-Diraby et al 2005; El-Diraby and Gill 2006).

Although current Semantic Web-related projects have an impact on construction product ontology modelling, such as bcTaxonomy and ifcXML, and process ontology modelling (El-Diraby, 2003 and Mutis et al, 2005), they are seldom concerned with ontological information in documents for both product and process models. Also, they have not developed any mechanisms for semantic-aware Web Services nor systematically integrated independently-developed ontologies and software/applications.

Pan et al (2004) concluded related researches and indicated potential Semantic Web applications in the construction industry as follows:

(1) Knowledge Management

In a large scale construction projects, there are many partners. Each partner has its own knowledge management system. To work together, they co-operate with each other using the common database of the project as well as sharing their personal databases. Currently most companies write databases in their own format. Meanwhile, a company may handle many projects at any one time. Since sharing

data among various databases is troublesome, one resolution technology is the Semantic Web. Content-driven semantic knowledge management makes the machine understand the information as well as human beings. So some inference and negotiation can be carried out automatically. This is both time and cost effective and bridges the gap of location, time zone, and language. For this purpose, the e-COGNOS project (2003) developed an ontology. A vision on the Semantic Web-based knowledge management (see Figure 2-10) was proposed by Anumba et al (2003), which provided enhanced access to construction knowledge and services. Further work is processed in this research.

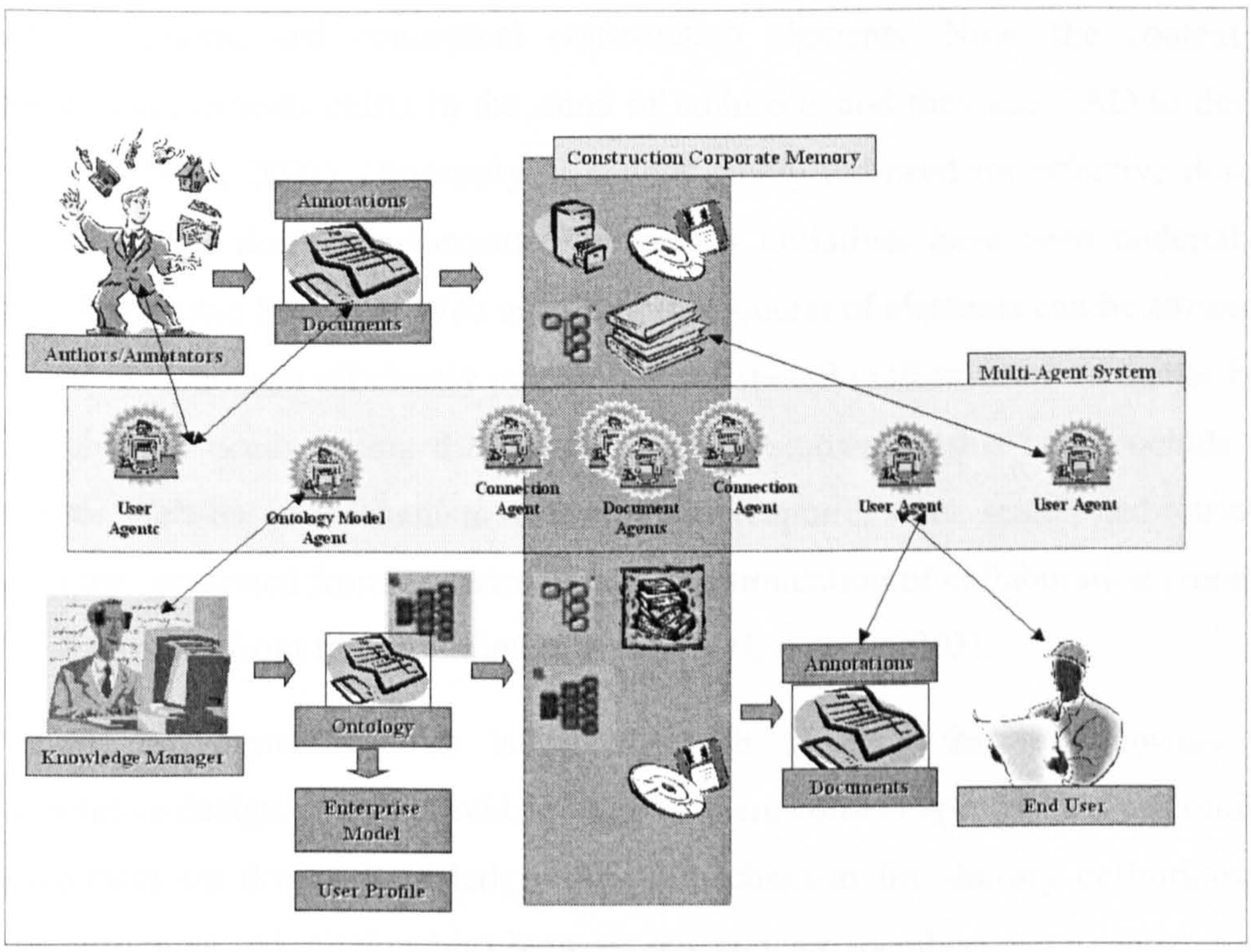


Figure 2-10: Construction Knowledge Management Using the Semantic Web (Anumba et al, 2003)

(2) Collaborative Design

Data and information generated at the design stage of a construction project reflect the tacit design knowledge. Data could be structured or unstructured. Client's brief concepts, the architect' briefing notes and sketches are mainly informal and not well structured. The final drawings and reports are structured. In the conceptual design, before a common agreement of the project comes out, the client and the architects may use different terms to define an item and many conceptual construction

elements are alternatives in a design. For instance, the architect can use either door or window for ventilation issues. Meanwhile in the detailed design, the same item may have different characters according to its context. For example, the installation requirements of an Ø10 mm bolt are different if it is fixed in a steel beam rather than on an entrance door. The difference leads to the fact that the bolts are different items in the project information management system. However current information management systems cannot distinguish them before installation. In the Semantic Web, they will be identified by their context when they first enter into the system.

The architect needs to consider design rules, functional requirements, economic and legal restrictions, and conceptual construction elements. Now, the context of construction elements exists in the mind of architects and they use CAD to design (Kraft and Nagl, 2003). Obviously, it cannot satisfy the need for effective design. To improve the design performance, numerous initiatives have been undertaken. Using the flexible Semantic Web ontology, the context of elements can be attributed to them. So users can efficiently manipulate the specific information no matter how many changes occur during the design. Primary studies on this issue include the Semantic Web-based mechanism to manipulate (capture, store, search and retrieve) knowledge generated from experiences and the simulation of collaboration amongst the stakeholders from the beginning of a project (Lai et al, 2003).

In short, the Semantic Web is an approach to meet the requirements for collaborative design. It will provide content-driven, context-specific and customised management on design knowledge, and a mechanism for sharing definitions of terms, resources and relationships between design team members.

(3) On-line Procurement

The open market brings global bidding to construction projects but can also bring problems. One of them is fuzzy descriptions of tender's requirements and productions, which lead to redundant search results for a query. On the contrary of information overload, a less relative one can be reached because countries use private standards for products and local procurement procedure. What's more, it is difficult to confirm bidder's qualification if the bidding is made online.

Europe E-Construction project has proposed a standard vocabulary, bcXML (2003), for building materials and set up product catalogues which provided on-line

directory service to users (see Figure 2-11). Upon the standard resource description framework and flexible ontology links, to add, edit or remove elements of the system and share information among systems would be easier to bidders.

Property	Value	Unit
country of origin		
availability		
delivery time		day
catalogue code		
european article number		
description		
product line		
short name		
trade-name		
list price		EUR (EUR)
uniform resource locator		
endurance		
operational durability		

Object: Material

Property	Value	Unit
country of origin		
availability		
delivery time		day
catalogue code		
european article number		
description		
product line		
short name		
trade-name		
list price		EUR (EUR)
uniform resource locator		
endurance		
operational durability		
has binding material		
has plasticizer		
has binding accelerator/retarder		
volumic mass		
compressive strength		
environmental load		
plasticity		
splitting tensile strength		
air content ratio		
binder ratio		
binder/water ratio		
has granular material		
has water		
has filling material		

Object: Concrete

Figure 2-11: Properties of Objects Material and Concrete, Generated by the Taxonomy Server (bcXML, 2003)

Beyond these services, the Semantic Web has the potential to retrieve related but indirect information by automatic reasoning using given rules. The most important, it provides effective search engine and personalised services according to tenders' claim. This search is a content-driven one, so it retrieves related information and filters unrelated ones automatically. Furthermore, digital signatures can be used to assure the Web of Trust. It uses various levels of trust showing how much the stakeholder is to believe.

For example, to construct a cable bridge, strands of cable are used. The contractor can instruct his Semantic Web agent to obtain through his Web browser, which may be fixed, handheld or wireless equipment, such as the PDA, laptop, desktop and mobile phone. The agent promptly retrieves information about Ø15.7 mm first class steel cable from the producer's agent, looks up several lists of providers, and checks for the ones meeting the project requirements and least costly including both the material price and transportation fee, serving with excellent or very good trusted rating. From the short list, it looks for a match between available volume of production and acceptable price, which are provided by the agents of individual providers through their Web sites. Using keywords with semantics or meaning, the search engine can find most relative terms and provide them to the agent through the Semantic Web. And the agent can also verify the qualification of the providers online. After computing, the agent will present the providers with a bidding plan. If the contractor doesn't like it for reasons such as a provider did few projects using this type of cable, the agent can redo the search with stricter preferences about provider's experience, and will present a new list. Sometimes, there will be warning notes such as the provider is a Korea company. The production standard of this type of cable is a little different from that in UK. The contractor will review the difference and decide that it is acceptable. In this process, the agent works well no matter the provider and the contractor use different languages, locate in different countries and can't meet for the time-lag and geographical distribution. On the other hand, the construction material or service providers can access more project information without laboured search and filter. Semantic Web brokers can identify related projects for them. The bidding process will be simpler and more accessible both by humans and computers because information is written in the computer processable languages, XML(S), RDF(S) and ontologies.

(4) On Site Information Management

Construction activity is a dynamic and complex one. Great efforts have been made to handle the changes and make real-time decisions. However it still lacks technical supports to dynamically synthesize contents, provide context-aware services, and real-time access information from multiple resources. Opposite to high volumes of information, how to deliver the right one to the right person on time is still under consideration.

Another issue is with the progress of construction, team members need to exchange information and negotiate issues together. A face-to-face meeting is the most popular approach, but sometimes, it is impossible, expensive, difficult or inconvenient to aggregate members together. Then the progress will be delayed.

With the Semantic Web, the construction team members' practices will be facilitated. The Web Services are provided on a need basis. The on-site information distribution becomes easer. That allows the project team members to access a wide rage of related information from technical drawings to specifications, even use wireless receiver, which greatly improve the efficiency of real-time information management. Aziz et al proposed an implementation scenario (2004) and deployment architecture (2006) for Semantic Web-based construction collaboration with wireless services (Figure 2-12).

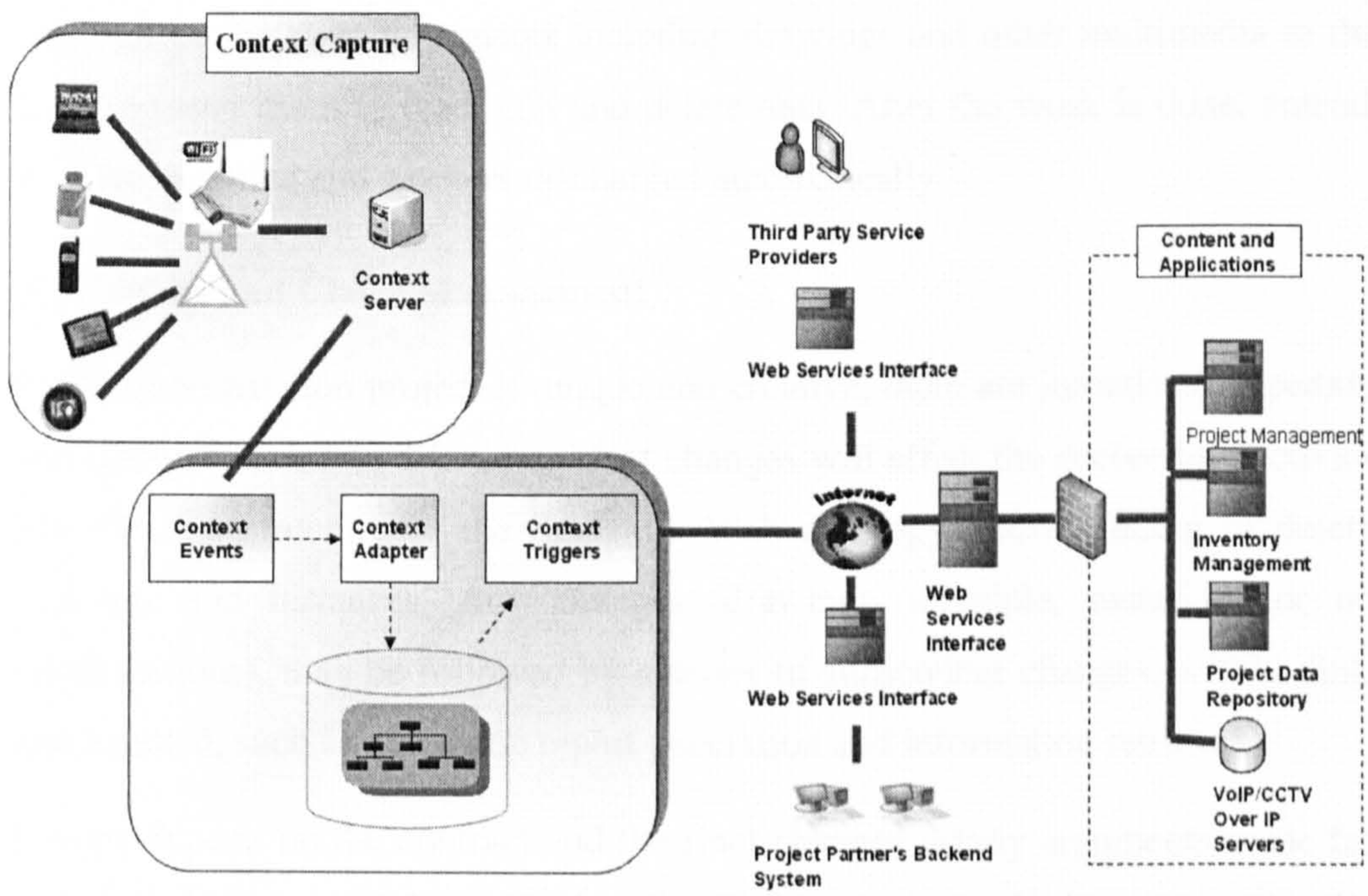


Figure 2-12: Deployment Architecture of Context-aware Services for Mobile Computing in the Construction Industry (Aziz et al, 2006)

(5) Communication and Collaboration

The Semantic Web provides a framework for sharing terminologies and data repositories between project partners. This enhances the ability of information management systems to automatically manipulate information. Then communication will be facilitated and made more effective (Aziz et al, 2004).

For example, partners develop and integrate the annotations with data (such as documents) in the knowledge management systems. One day, there will be three work-groups to work in a site, an electrical engineer, two plasterers and four carpenters. The electrical engineer will check the possibility to install an electric engine. The plasterers will finish the ground with mortar. The carpenters will install the wallboards. Using intelligent collaboration application, the on-site agent draws the requirements of each task from project database and the availability of workers from groups' databases. Then it works out a work plan considering the restrictions of space, time, material supply, and etc. The work plan is delivered to the groups' agent and gets feedback of whether they agree with this plan or not. If not, it will be revised to satisfy the additional requirement. Communication between agents will be finished in several seconds if there is no need for human participation. When work groups enter the site, if their trustworthiness is assured, the knowledge system will deliver the right documents including drawings and other multimedia to them and empower them to read, add and delete data. After the work is done, amended data are recorded and workers discharged automatically.

(6) Change and Claim Management

Since a construction project is unique and creative, there are initiatives, uncertainty and changes during its life time. Most changes will affect the succeeding activities. The flexible structure of the Semantic Web ontology allows adding or deleting concepts and instances. Any changes (drawings, schedule, materials, or non-confirmations), may be followed by a series of subsequent changes, will be linked and handled, such as automatic report generation and information retrieval.

Claims depend on the contract and the final changes. Many arguments come from lacking efficient communication and effective proofs. In the Semantic Web, information is marked-up and linked to others. Its evolution is also traceable. A computer can understand the basic concepts, such as subclass and inverse. Rules and logical principles could be stated and the computer could use these rules to reason and infer concepts and instances. That also enables it to attach proofs to support a claim.

To summarise, the Semantic Web enables information to be processed by machines not just for display purposes, but for automation, integration and reuse of data across various applications. The Semantic Web provides intelligent KM and WS to many knowledge-intensive areas.

2.3.3 The Key Features, Enablers, Barriers, and Potential Benefits from Adopting the Semantic Web

With a set of mark-up to the information sources on the World Wide Web, the Semantic Web will make data and services more accessible to computers and useful to people. This feature enabled several initial industrial applications of the Semantic Web, which revealed that the Semantic Web can improve the efficiency of Knowledge Management and Web Services in various industries. The machine-processable semantics of Web sources will bring numerous benefits:

- Manipulating information by both humans and machine: This will enhance the efficiency of processing information.
- Keeping data simplicity and logical consistency: Maintaining the links between related data will ensure that up-to-date information is available to all project partners.
- Interoperating independent databases and information sharing between users: Communication can be easier and faster, as, with the Semantic Web, both humans and computer agents participate in processing information in a peer-to-peer environment. Support can be provided to an agent-based organisation, which will lead to several additional benefits (Ren and Anumba, 2004).
- Providing customised services to users' requirements: Queries can be content and context-driven in line with individual requirements.
- Enabling more expressions of data: Links and annotations of the data in the Semantic Web make it possible to process unstructured data. Therefore, multimedia data can also be available using Semantic Web ontology (Schreiber et al, 2001). Graphics, sounds, and visual documents can be recorded, understood, retrieved, compared, and deleted by computers as well as humans. This suggests that information management can be made more visual.

- Operating across many applications (e.g. PDAs): Support can be provided to mobile users remote operations. This is expected to be a key feature of next-generation construction.
- Supporting the development of advanced knowledge management systems at various levels.

The Semantic Web application has not yet reached its full potential because the development of Semantic Web is still at a very early stage. Some of its components and associated editing tools are being deployed, and many are coming in the next years. The initial industrial applications greatly facilitate the Semantic Web's development. Many Semantic Web-related software and methodologies are developed in those projects. It is expected that its application in the construction industry can also kept a win-win relationship with the Semantic Web's development.

2.4 SEMANTIC WEB SOFTWARE

Many tools are available to develop Semantic Web applications. Some software, which edit annotations and ontologies, have been mentioned in the review of Semantic Web applications. The RDF/XML Parser, Ontology Editor and Ontology Reasoner are major types of Semantic Web tools. In this section, they are reviewed and the ones used in this research are highlighted.

2.4.1 RDF/XML Parser

The existence of standards for describing data (RDF) and data attributes (RDF Schema) enables the development of a set of readily available tools to read and exploit data from multiple sources. RDF parsers work as parsing different RDF serialisations, accessing RDF triples via programming interfaces or queries, and providing basic operations with the RDF triples. Since the RDF data are coded in XML, RDF Parsers are able to process XML during parsing RDF syntaxes. There are many RDF Parsers and these are summarised in Table 2-2, where they are sorted by the computing language used.

In large projects, such as On-To-Knowledge, the RDF parser (Sesame), ontology editor (OntoEdit) and end-user application (Ontoshare) were integrated into an application environment as shown in Figure 2-6. Most recent ontology editors

incorporate basic RDF editing functions, which allow users to specify the attributes and values of a concept.

Table 2-2: RDF Parsers Review List (McBride et al, 2003)

Computer Language	RDF Parsers
Java	Sesame, Jena, ICS-FORTH RDFSuite, IsaViz,
Python	Metalog, Redfoot RDF framework, W3C Semantic Web Area for Play, 4Suite 4RDF Python library
Perl	RDFStore, CARA
LISP	Wilbur
C	Redland RDF Application Framework, Raptor, RDFDB
C#	Drive, EulerSharp
Tcl/ Tk	XWMF
PHP	RAP

2.4.2 Ontology Editors

An ontology editor is a software tool for editing ontologies. Many early ontology editors provide explicit support for particular knowledge engineering methodologies. Recent tools have an increasing tendency to support common upper level ontologies. Four major editors are introduced as follows:

(1) OilEd

OilEd (Figure 2-13) is a simple ontology editor developed at the University of Manchester. OilEd allows the user to build ontologies, use the FaCT reasoner to check the consistency of ontologies, add implicit subClassOf relations, and export ontologies in a number of formats including both OIL-RDF and DAML-RDF (OilEd, 2002). OilEd was originally designed as an OIL/DAML+OIL editor, but now supports OWL. It will read and render OWL in RDF/XML.

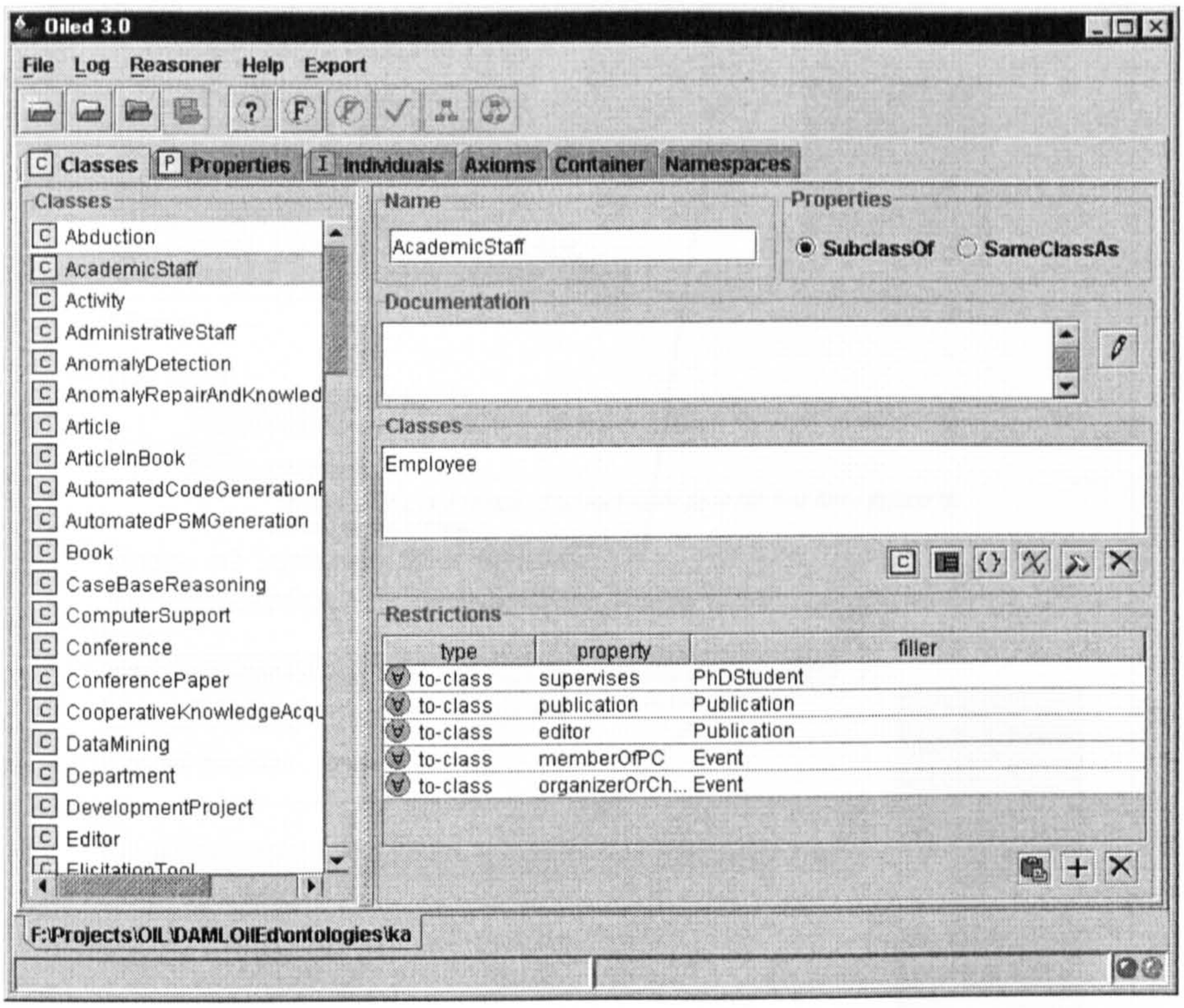


Figure 2-13: OilEd Ontology Editor Example (Bechhofer et al, 2001)

(2) Ontolingua

Ontolingua (2003) was developed by the Knowledge System Laboratory (KSL) at Stanford University. It is a set of tools and services that support the sharing and reuse of ontologies among AI (Artificial Intelligence) systems, which included Chimaera (a browser to access a library of ontologies and resolve name conflicts in a Knowledge Base), an ontology editor (used to create and browse ontologies), Webster (a HTTP gateway to a Webster server), and translators (used to export ontologies in diverse languages, such as Ontolingua and Prolog). Ontolingua is an on-line ontology editor free to registered members. However, it does not provide any downloadable version. Figures 2-14 and 2-15 are screenshots from the Ontolingua Ontology Editor for the Vehicles Ontology (Farquhar et al, 1997).

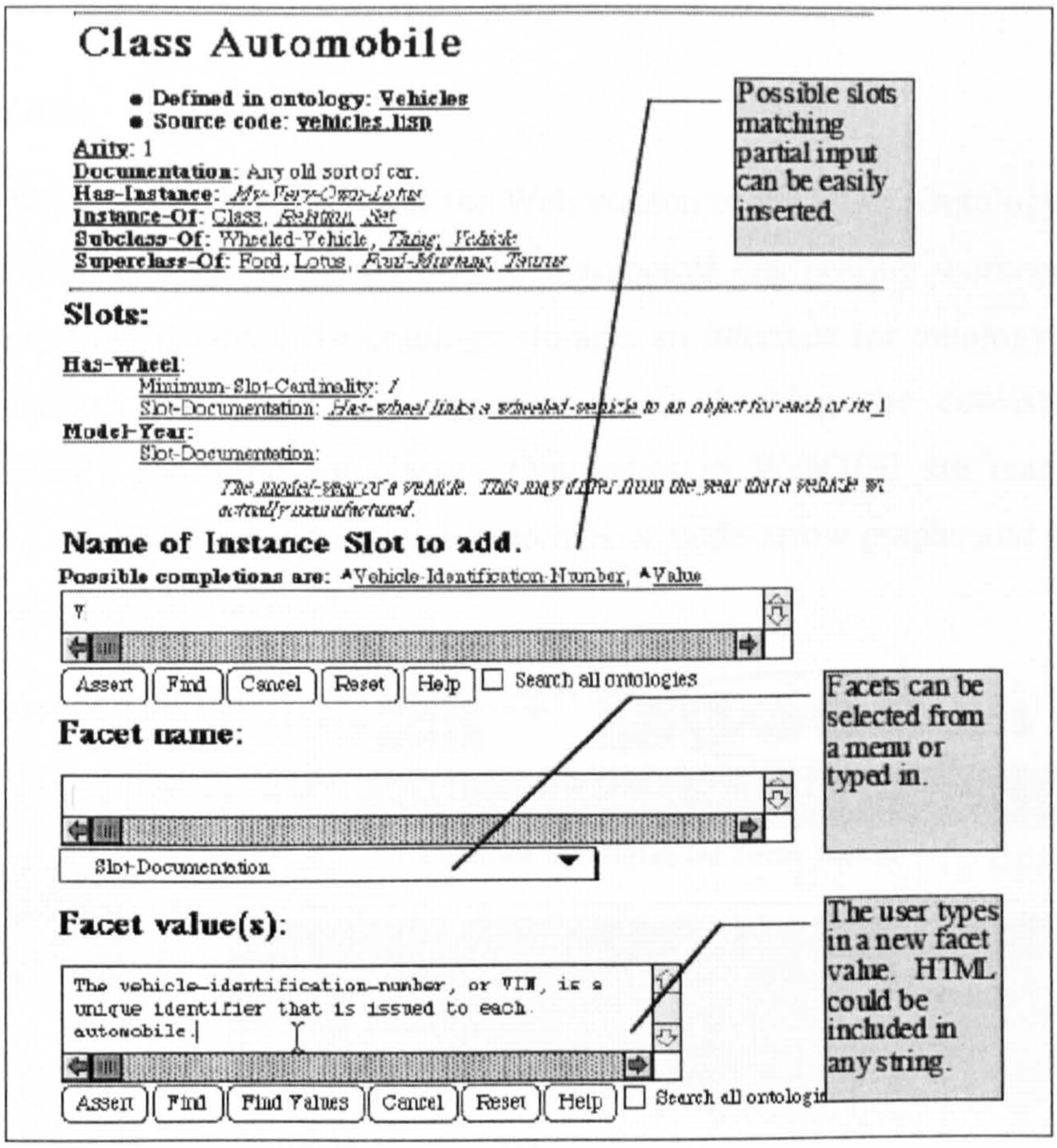


Figure 2-14: Class Editing in Ontolingua (Farquhar et al, 1997)

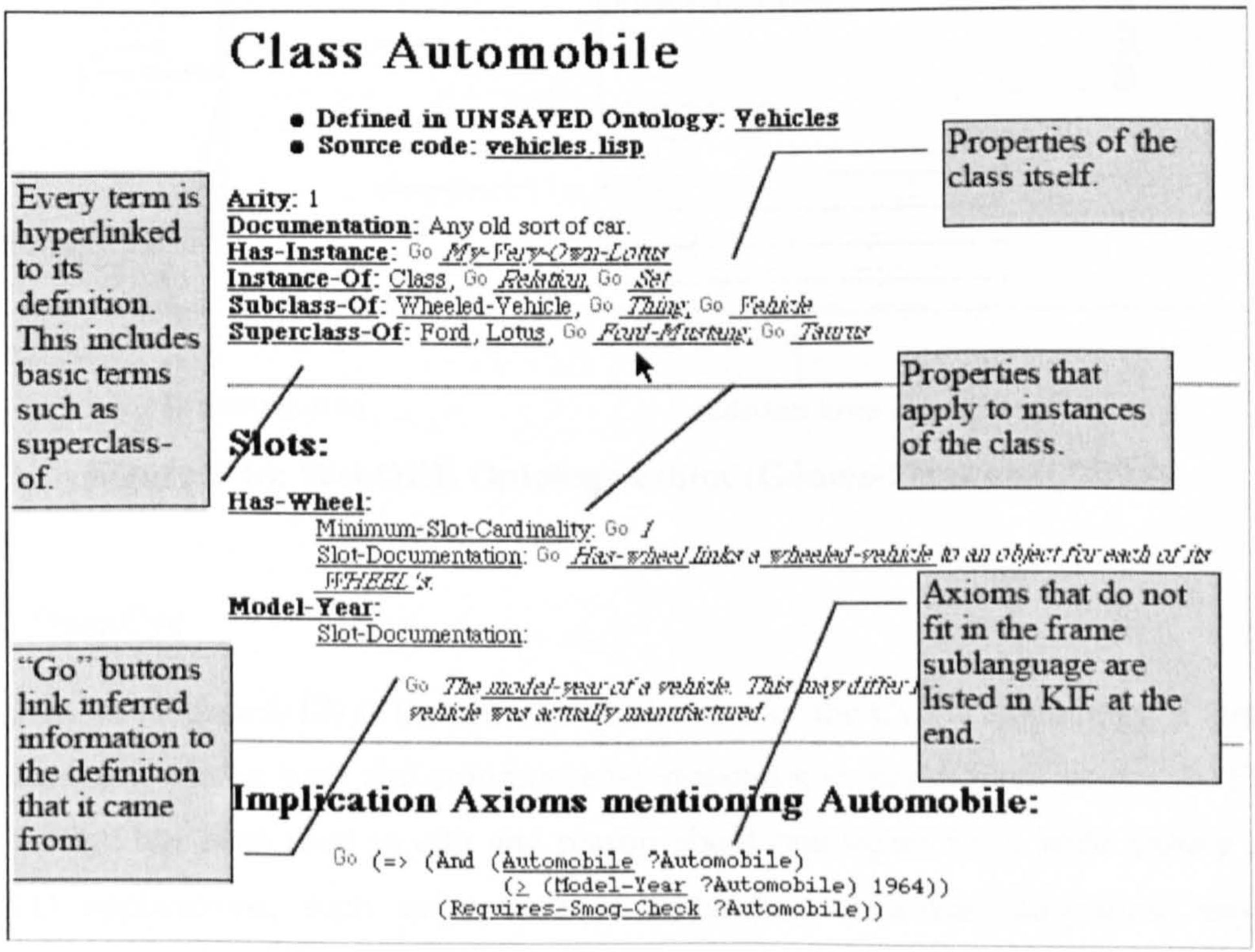


Figure 2-15: Class Definition Browser in Ontolingua (Farquhar et al, 1997)

(3) WebODE

WebODE (Figure 2-16) (2003) was the Web version of the ODE (Ontology Design Environment). It is an integrated on-line ontological engineering workbench that provides an SQL database for ontology storage, an interface for ontology editing, and a reasoner for classifying synonyms and checking the consistency of type/cardinality constraints of classes. Ontologies in WebODE are manipulated online and can be visualised in tree hierarchies or node-arrow graphs and coded in various ontology languages.

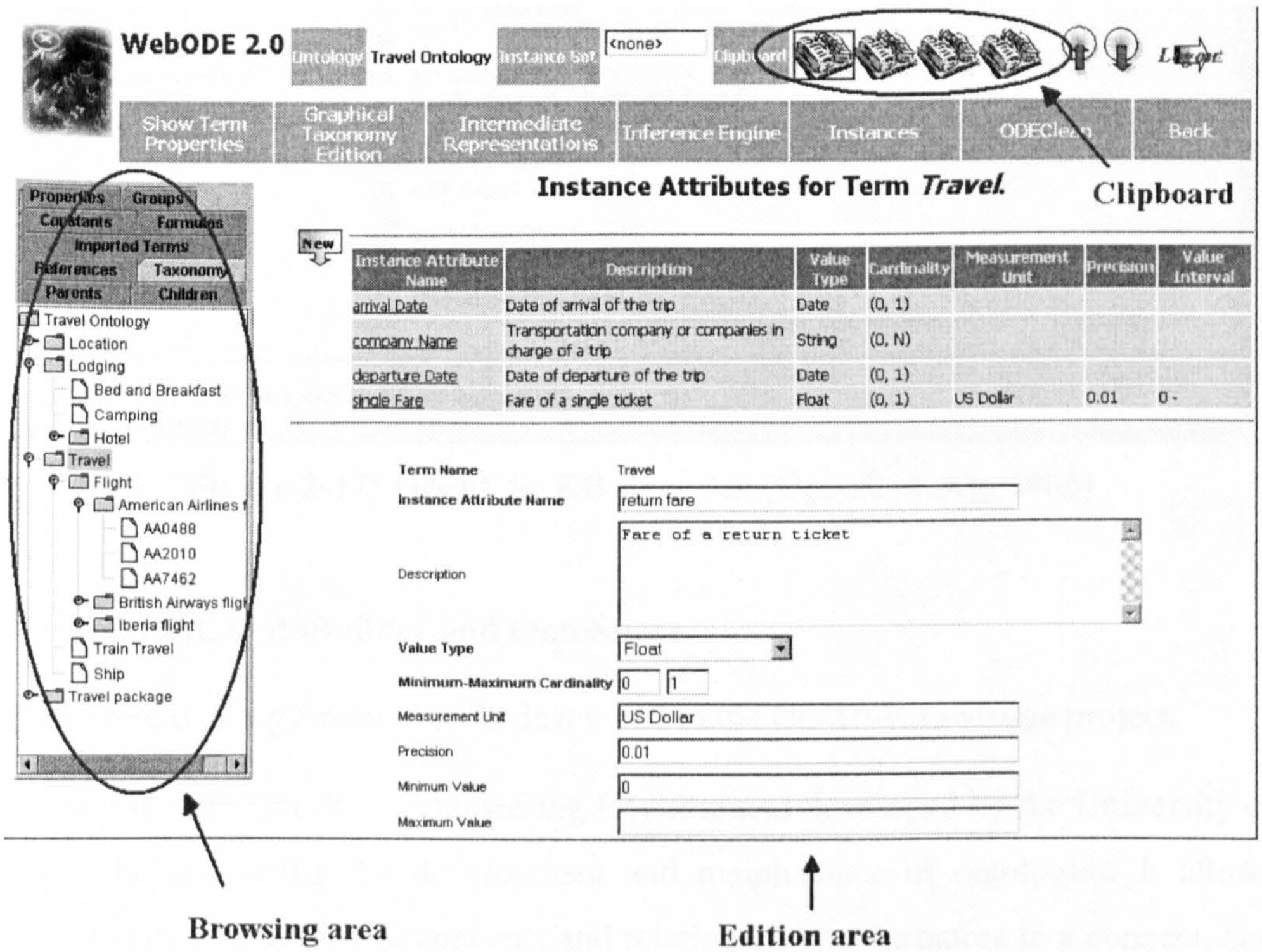


Figure 2-16: WebODE Ontology Editor (Gómez-Pérez et al, 2003)

(4) OpenCyc

OpenCyc (Figure 2-17) is the open source version of the Cyc(r) technology, a large general knowledge base and commonsense reasoning engine (OpenCyc.org, 2005). OpenCyc has been used to edit and reason about ontologies for a wide variety of R&D applications, such as: speech understanding, database integration, rapid

development of ontology, email prioritising, routing, summarisation, and annotation to name a few.

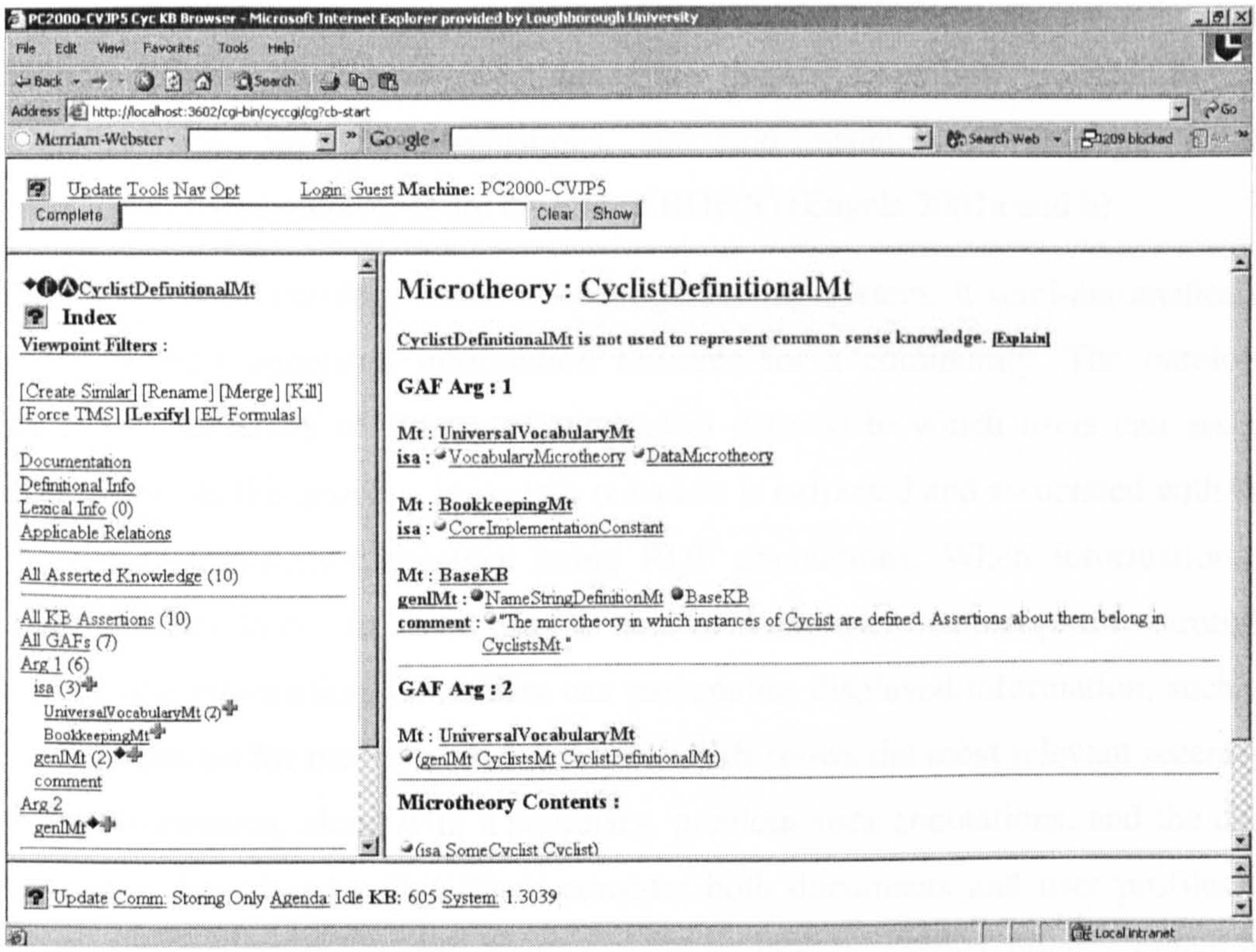


Figure 2-17: OpenCyc KB Browser (OpenCyc.org, 2005)

(5) OntoEdit, OntoBuilder and OntoShare

They were ontology-related tools developed in the On-To-Knowledge project.

OntoEdit is an Ontology Engineering Environment developed by the University of Karlsruhe supporting the development and maintenance of ontologies. It allows users to edit a hierarchy of concepts and relations, create instances to a concept, and grant a concept several names, which essentially is a way to define synonyms for that concept (Sure and Studer, 2001).

CORPORUM-OntoBuilder is composed of two main modules: CORPORUM-OntoExtract and CORPORUM-OntoWrapper. They can tackle both structured and unstructured documents. OntoWrapper is used to deal with structured documents and uses screen-scraping and business rules to extract information from known places on specific sites (e.g. names, email addresses and telephone numbers). OntoExtract is used to deal with unstructured documents. It extracts initial

ontologies/taxonomies from natural language on Web pages. OntoExtract can conduct semantic analysis of the content of Web pages. It is able to provide initial ontologies/taxonomies, refine existing ontologies (to include more concepts), find relations between key terms in documents, and thereby relate business areas to each other or allow for new associations, and find instances of concepts within documents. These ontologies are created in RDF(S) (Engels 2001a and b).

OntoShare is an ontology-based knowledge sharing system. It semi-automatically builds an RDF-annotated information resource for a community. The ontology specifies a hierarchy of concepts (ontological classes) to which users can assign information. In this process, important metadata is extracted and associated with the community information resource using RDF annotations. When information is shared in OntoShare, an email alert is sent to those users whose profile strongly matches the information. OntoShare can personalise displayed information, such as the "Documents for me" to a specific user, which shows the most relevant recently-stored information, along with a summary, previous user annotations, and the date of sharing. Searches by OntoShare consider both documents and user profiles to match a given query. In this way, a user is able to contact other community members whose profile matches a given topic, thereby encouraging possible tacit knowledge exchange. Ontoshare can support a degree of ontology evolution based on usage of the system. The system also attempts to modify the ontology to better reflect the user's conceptualisation (Davies et al, 2003).

(6) Protégé

Protégé (Figures 2-18 and 2-19) is a Java ontology editor and a knowledge-base editor, which provides an extensible architecture for the creation of customised knowledge-based applications (Protégé, 2006). The first Protégé application was built for knowledge-based systems in 1987. After years of evolution, the latest version, Protégé 3.2 Beta, can be run on various platforms and has been used by hundreds of individuals and research groups. Protégé 3.2 Beta has two editing environments: Protégé-Frames and Protégé-OWL. The Protégé-Frames editor enables users to build and populate ontologies that are frame-based, in accordance with the Open Knowledge Base Connectivity protocol (OKBC). In this model, the ontology consists of a set of classes organised in a hierarchy to represent a domain's

concepts, a set of slots associated to classes to describe their properties and relationships, and a set of instances of those classes - individual exemplars of the concepts that hold specific values for their properties. The Protégé-OWL editor enables users to build ontologies for the Semantic Web, in particular in the W3C's Web Ontology Language (OWL). OWL ontology includes descriptions of classes, properties and their instances, which are called 'individuals' in Protégé-OWL. Given such ontology, the OWL formal semantics specifies how to derive its logical consequences, i.e. facts not literally present in the ontology, but entailed by the semantics. These entailments may be based on a single document or multiple distributed documents that have been combined using defined OWL mechanisms (W3C 2004). Many plug-ins are available as extensions to the core Protégé system, such as visualising an ontology by OWLViz plug-in (Figure 2-19) and merging ontologies by PROMP plug-in.

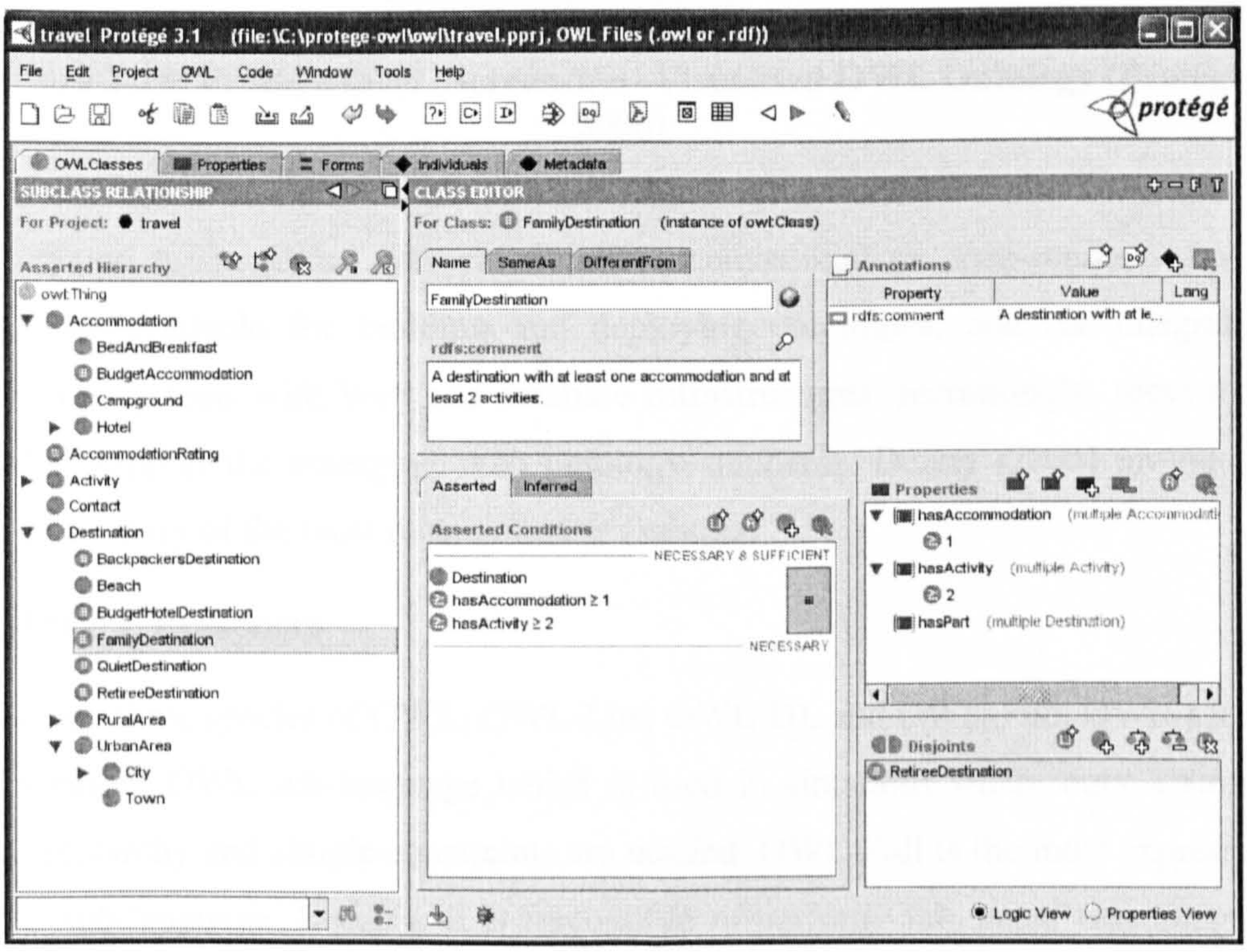


Figure 2-18: Protégé-OWL Screenshot - OWLClasses View (Protégé, 2006)

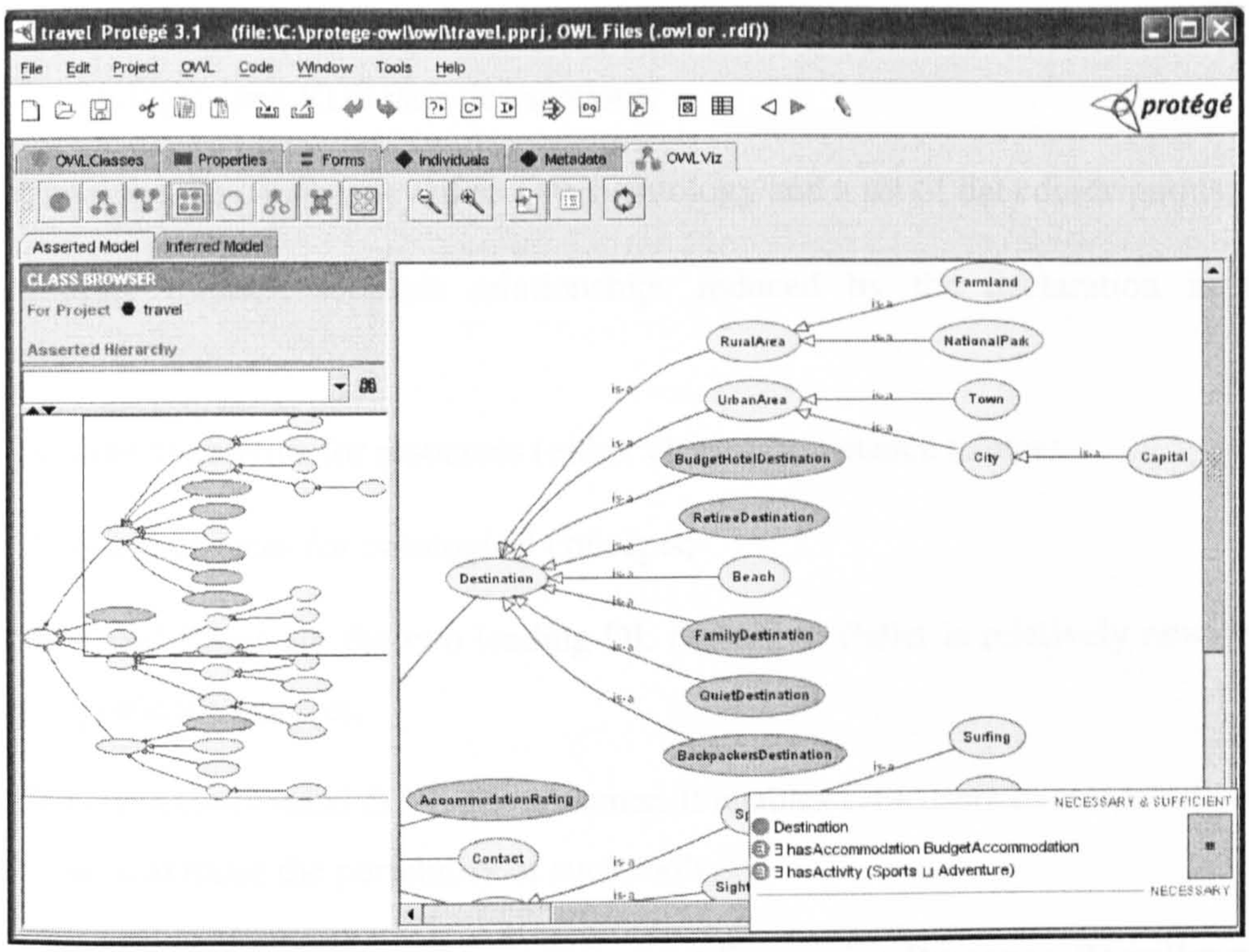


Figure 2-19: Protégé-OWL Screenshot- Visualised OWL Ontology (Protégé, 2006)

In addition to the above editors, numerous commercial or open-source software tools are available for building and deploying ontologies, and for integrating inference systems with Web and database infrastructures. Increasingly, these tools directly support the emerging Web ontology standards. Denny (2004) provides a good summary of the most recent ontology editors.

2.4.3 OWL Reasoners

There are three species of OWL: OWL-Lite, OWL-DL and OWL-Full. OWL-Lite is the simplest OWL sub-language which is used in situations where only a simple class hierarchy and simple constraints are needed. OWL-Full is the most expressive OWL sub-language. Because it is impossible to perform automated reasoning on OWL-Full ontologies, OWL-Full is only used in situations where very high expressiveness is more important than being able to guarantee the computational completeness of a language. The expressiveness of OWL-DL (Description Logic) falls between OWL-Lite and OWL-Full. Ontologies described using OWL-DL are computable by a reasoner. Available OWL-DL reasoners include FaCT++ (2006),

Racer (2006) and Pellet (2006). These reasoners provide the following services for OWL ontologies and RDF data descriptions:

- Checking the consistency of an OWL ontology and a set of data descriptions;
- Finding implicit subclass relationships induced by the declaration in the ontology;
- Finding synonyms for resources (either classes or instance names);
- Finding instances for ontological concepts;

FaCT++ and Racer are the two leading DL reasoners. Pellet is relatively new with some significant features.

FaCT++ is a C++-based OWL-DL reasoner. It enables C++ users to create software tools and maximise the portability of such tools.

Racer stands for Renamed ABox and Concept Expression Reasoner. RacerPro was the commercial name of the software, which can be used as both a Semantic Web reasoning system and an information repository. It also enabled an HTTP client for retrieving imported resources from the Web. Multiple resources can be imported into one ontology. Racer is rather easy to use through a HTTP based DIG interface (also known as DIG, which is a standard XML-based language developed by DL Implementation Group) for reasoning with OWL (there is also a file and TCP interface). A number of ontology editors (including OilEd and Protégé) can use the DIG interface to communicate with Racer. An OWL-QL (Query Language) query processing system was available for RacerPro users to query extensional information from OWL documents (OWL instances and their interrelationships). The first implementation of the Semantic Web Rule Language (SWRL) was provided with the latest version of Racer (Racer Systems GmbH & Co KG, 2005).

Pellet is an OWL-DL reasoner based on the tableaux algorithms developed for expressive Description Logics. It is the only DL reasoner that can support the full expressivity of OWL-DL including reasoning about nominals (enumerated classes).

Generally speaking, compared to the RDF parsers and ontology editors, OWL reasoners are relatively under developed.

2.5 SUMMARY

The Semantic Web is proposed as the next generation of World Wide Web. It combines several techniques (URI, XML(S), RDF(S), Ontology, RIF and PML) enabling the meaning of information to be explicit and processable by computers. Many software tools have been developed to facilitate information annotation, ontology editing and description logic reasoning. Many of these have been used in projects in specific domains. For example, Protégé was used in medicine and biomedical ontology. Most applications of the Semantic Web in the construction sector are high level frameworks and fragmented ontologies for specific domains. They neither identify the ontology of information in documents for whole construction project processes nor support semantic-based information management services. A systematic study on the Semantic Web-based information management system which integrates existing research outputs is needed. Referring to the experiences of Semantic Web industrial applications, the features of the Semantic Web and the benefits from adopting it are summarised. Its potential application areas in the construction industry are highlighted and the advantages of a Semantic Web-based information management system over traditional information management approaches for construction projects are discussed further in Chapter 3.

CHAPTER 3 - CONSTRUCTION PROJECT INFORMATION MANAGEMENT

3.1 INTRODUCTION

This chapter provides an overview of theories and technologies related to construction project information management. Firstly, it introduces the subject of information management, its concept and technology. Then, Information Technology (IT) related problems of fragmentation and data incompatibility in the construction industry are investigated with a list of the Semantic Web opportunities. Existing information classification mechanisms in construction sectors are examined. The world-wide standardisation efforts are reviewed in five projects: the STEP (STandard for the Exchange of Product model data) and ISO 12006 (framework for organisation of building construction information) by the International Standards Organisation (ISO), the LexiCon (the development of a taxonomy for building and construction industry) by the European eConstruct project, the IFC (Industry Foundation Classes) by the International Alliance for Interoperability (IAI), and the UniClass (Unified Classification for the construction industry) developed by the National Building Specification (NBS) on behalf of the Construction Industry Project Information Committee (CPIC), followed by a discussion of construction information taxonomy used in this thesis.

3.2 DEFINITION

3.2.1 Information Management

The importance of information has long been recognised and information is considered as a resource today, which is an asset to a person or an organisation (Macevičiūtė and Wilson, 2002). Since diverse sources of information exist in the real world, considerable efforts have been put on exploring effective approaches to manage information, which are called information management methods. In this thesis, we adopt the definition by Wikipedia (2006 a) to define the information management as "the handling of information acquired by one or many disparate sources in a way that optimizes access by all who have a share in that information or a right to that information". Information handling activities include information

receiving, generating, administering, monitoring, storing, securing and disposing (Aspinall, 2004).

3.2.2 Construction Project Information

Construction projects generate, absorb, and transfer significant quantities of information. Construction project information management means the handling of project information acquired by construction project partners in a way that enables the information handling activities to be more efficient and effective.

Construction project information is complex because the construction industry is characterised by a high level of fragmentation. This fragmentation is considered from both geographical and functional perspectives (Chan and Leung, 2004). The former results from the geographical dispersion of project partners (horizontal fragmentation), whose collaboration is transient. The latter fragmentation perspective relates to the various disciplines involved in the project throughout the entire construction process (vertical fragmentation) (Howard et al, 1989). Fragmentation creates difficulties in collaboration and co-ordination between the various partners. In particular, it is often expensive and inconvenient to have face-to-face meetings to resolve problems when the partners are globally distributed. In addition, project partners may use different vocabularies. This leads to considerable potential for misunderstandings and miscommunication. Existing information management applications do not adequately address these fragmentation problems (refer to the review in Section 3.3) and new approaches facilitating collaboration and communications in project team members are in great need.

3.2.3 Information Management and Document Management

The major media for handling information among project partners in the construction industry is various kinds of documents, schedules, design drawings, procurement orders, contracts and meeting minutes to name a few (as shown in Figure 3-1). The construction project information management process is characterised by handling documents held by project partners located in different places. An inexact surveying record being used by an engineer or an outdated design file being used by a contractor can lead to huge economic loss or even project failure. Thus the key issue in information management in the construction industry is to manage project documents precisely, timely, accessibly and

effectively among diverse partners. However this is not easy due to the fragmentation nature of construction projects.

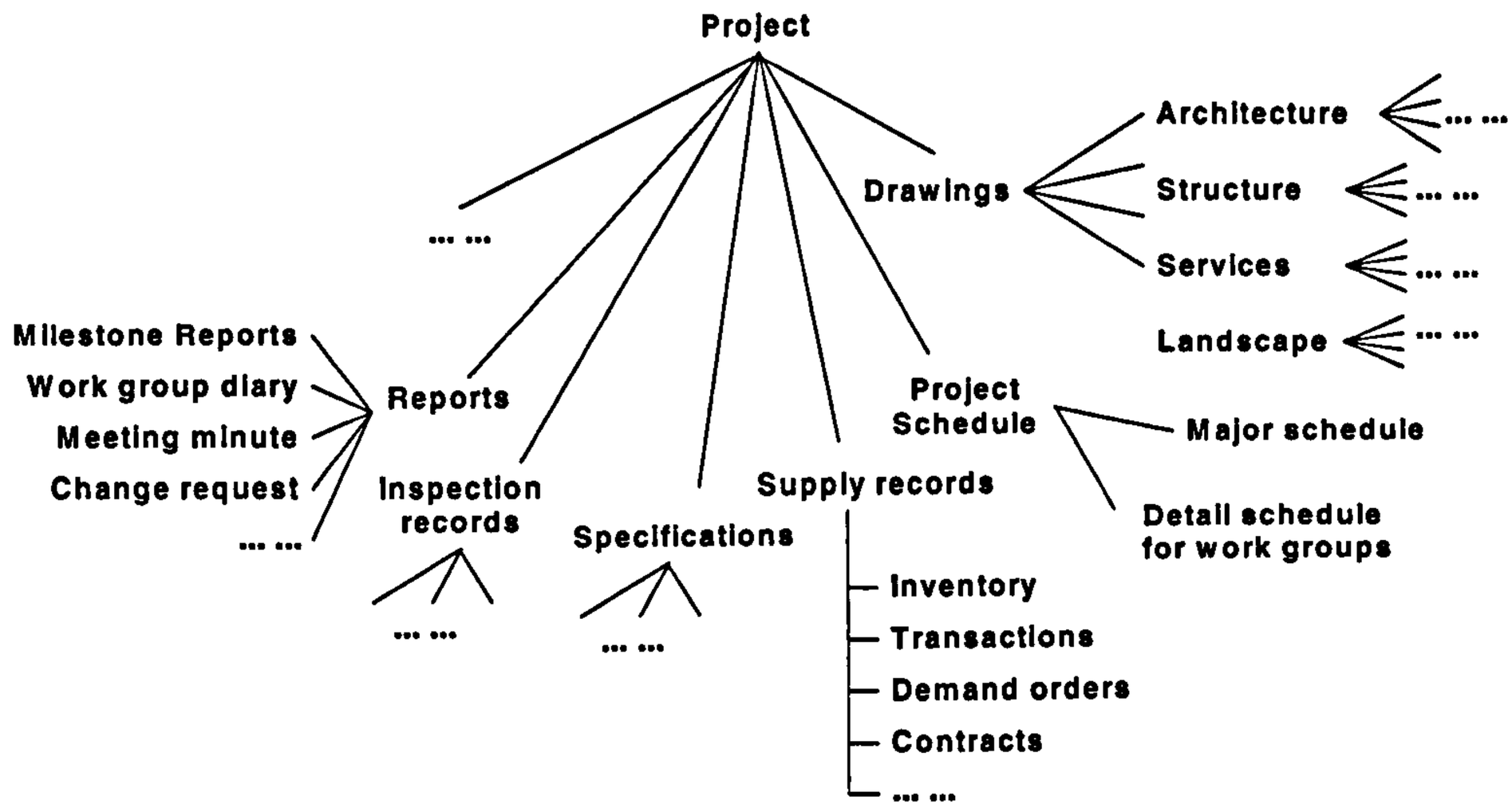


Figure 3-1: A Mass of Documents in a Construction Project

Many methods have been developed gradually to facilitate the management of information in a document level.

3.3 HISTORICAL PERSPECTIVE

The evolution of project information management methods can be categorised into three stages: manual information management stage, electronic information management stage and Web-based information management stage. The Semantic Web-based information management can be considered as the next stage after Web-based information management.

3.3.1 Paper-based Manual Information Management Stage

At this stage, it was people who completed all the information editing, recording, query, and retrieval tasks. If a person wanted to conduct a search, he/she needed to find out who was in charge of the issue and talk to them. The relevant person will think over the demand and identify where the relevant information was stored. If he/she knew, he/she will respond to the enquirer, either providing some documents or telling the enquirer where he/she can get the information. The process may be repeated until the person gets the right information. Some uncertainty exists in the

process caused by many factors, like asking the wrong person, misunderstanding of the request, or too much information to handle.

3.3.2 Electronic Information Management Stage

In the 1970's, electronic project information management tools using Information Technology (IT) emerged (Otjacques et al, 2003). Here, IT refers to technology, which deals with the use of electronic computers and computer software to convert, store, protect, process, transmit, and retrieve information (Wikipedia, 2006 b). Much information was managed in electronic format files when the volume of information increased with the quantity and scale of projects. Electronic information management systems enable people to handle much more files within limited time and space than ever. Because electronic files can easily be transferred to people and organisations for communication, IT is also known as Information and Communication(s) Technology (ICT) and Infocomm.

However, at this stage, the information exchange between different information repositories/sources (persons or organisations) was still processed by people. Although the efficiency of indexing information in a single source was improved, it is still a problem to identify the source of searched information and contact relevant people. Misunderstandings still exist in the communication between people. Furthermore, the data structures are often not compatible between independently developed information systems. It requires professionals to convert information from one data structure to another, and the cost of doubled manual input cannot be avoided (Emmerik, 2000).

3.3.3 Web-based Information Management Stage

By the 1990's, information was mainly conveyed by various electronic files and regularly disseminated across computers and other electronic devices. With the development of the World Wide Web, distributed information resources can be connected. Then electronic files were shared between project partners by Web-based information management software. Project partners can use websites to get real-time information, review the response to an RFI (Requests For Information), view updated plans, and track the progress of submittal approvals (Otjacques et al, 2003). At the Web-based information management stage, information was uploaded to the Internet and displayed to remote users, which enabled enquirers to search

information by themselves and obtain it in electronic format. Some projects used extranet as a private network to share information in a specific domain. Project Extranet (Wikipedia, 2005) is a "dedicated network with many features such as e-mail, message board, document repository, diary, up-coming event list, and contact list". Examples include Asite (2000), 4Projects (2001), and BIW (2001). These systems and associated tools provided a platform for globally distributed project team members to communicate and exchange documents. Although Internet technology strengthened the connectivity of distributed information sources, it only managed information in a document/textual level. Most Websites were designed to display documents to humans rather than to process information by computers. Using e-mail, CD or FTP, documents were disseminated in electronic format. However the diversity of electronic formats used in construction projects often leads to difficulties to automatically process information wrapped in documents. Some problems exist in the document management process, such as information overload, loss, and misunderstanding. Further effort is required to achieve semantic-aware document management.

3.4 CHALLENGES

The manual, electronic and Web-based information management systems supported construction project information management in various extent of satisfaction. However, because of the fragmented nature of construction projects information, there are still many challenges, such as information overload, loss, outdated, inconsistency and misunderstanding to be tackled. Their symptoms are as follows:

1) The semantics of information is hard to access. Project partners have to spend most time in reading the retrieved document to access its contents, then extracting the useful part and identifying if it meets the needs of a given query.

2) It is laborious to access related documents in diverse databases. For example, an architect can get the blueprint of a floor from the drawing database, but cannot retrieve its design specification at the same time because it is in specification database. Another query to the specification database is required in this case. Furthermore, if the users are unfamiliar with the database structure/language, they have to spend more time on a specific search.

3) The keyword-based search, which is prevalent in most electronic information management systems, is imprecise. The keywords in documents are often pre-defined while the searched terms are free-text. There are problems in matching the searched terms to document keywords because enquirers may use their own terms to express the meaning of those keywords. Therefore, the ability to choose an appropriate keyword in searching affects the efficiency of information retrieval. Although it is possible to search across multiple information repositories for documents containing the same keywords, there are still extensive homonym and synonym problems within and between various repositories. Thus, searches often return irrelevant information, or miss information when different terms with the same meaning are used, and fail to recognise the relationships between different pieces of information. For example, in a design team, everyone knows that a lift can also be called an elevator. However, computers do not know this and thus cannot return "elevator" information when the searched keyword is "lift".

4) Information/documents loss limits the efficiency of construction project information management. Information is wrapped in documents but its meaning and the relationships between information are not well annotated to documents. Therefore, during information dissemination and the retrieval process, related documents may be lost.

5) Information/documents which are out of date can lead to failure of a project. It is difficult to keep documents up-to-date for all potential users. The document creator and project manager are usually responsible for the documents distribution issue. However they normally have no or less-exhaustive user list. Therefore, someone may use the outdated documents, especially those getting the document from a third party, not directly from the document's creator. An outdated design file being used by a contractor can lead to huge economic loss or even project failure.

6) Information/documents overload annoys project partners. Information contained in documents is hardly accessed during the document management process. Electronic documents have become the primary media for handling project information between project partners. It is easy to upload an electronic document onto a Website and disseminate it by e-mail. Project partners may

receive piles of unnecessary information in an e-mail box or get irrelevant responses to their online query because computers cannot readily identify unrelated documents. It is time-consuming to filter irrelevant documents.

7) Information confusion or misunderstanding may occur because each project participant prefers to use his own terminology/technical jargon. Even within a discipline, different companies may use different terms to refer to the same object. The responsibility for handling various terminologies places a burden on relevant individuals and their ability to interpret terminology strongly depends on experience and knowledge.

8) It is hard to realise the automatic interaction between companies because most companies develop their own databases and operational systems independently (Löwnertz, 1998). The information in one system cannot be recognised by another system because different systems use different data structure, categorization, and vocabularies in their domain. So specialists who know both systems are expected to handle the inter-operation issue. It is inconvenient to resolve problems and reach an agreement when project partners use different languages, cannot meet together simultaneously, or have different perspectives on an issue.

9) It is hard to maintain the consistency of updated information throughout a variety of applications when information changes. Because they may be programmed independently, using different languages, file categories and data structures, it costs double manual work to input the changes into applications.

Enabling information contained in documents to be accessible, processable and interpretable by both humans and computers is an approach to meet the above challenges.

3.5 SEMANTIC WEB OPPORTUNITIES

The Semantic Web is considered as a new tool that can support next-generation information and knowledge management systems for the construction industry (Christiansson, 2003). It provides many good opportunities to meet the above challenges. Some important ways in which the Semantic Web can improve the status quo are briefly described below:

1) Content-based information/knowledge sharing and management;

2) Context-specific information management services for project partners;

3) Automatic inter-operation between diverse information sources/applications;

The Semantic Web provided experiences in overcoming information management obstacles between construction project partners. More about the concepts and industrial applications of the Semantic Web have been covered in Chapter 2. This chapter focuses on the attributes and interpretation of project information contained in documents, so as to process the information in a Semantic Web environment.

3.6 INFORMATION IN CONSTRUCTION PROJECT DOCUMENTS

In this research, construction project information was defined/annotated by ontological concepts (associated with properties), ontologies (relationships between concepts/properties) and rules (constraints to the ontologies). Before developing an annotation model and ontology, the following tasks were processed:

1) Identifying the types of project documents. Analyzing the semantic content of existing project documents and the types of information contained therein;

2) Reviewing available construction project information taxonomies, frameworks and definition languages. Major projects attempting to standardise the classification mechanism for project information were discussed;

3) Identification of how project information types needed to be developed for use in a Semantic Web environment.

3.6.1 Document Types

There is no standard to categorise project documents for all companies. Each company's information management system adopts its own category. For example, an architecture drawing for project ABC may be classified into "project ABC" folder in an architect's system, while a copy of the drawing was classified in the "drawing" folder in the project manager's system. No link existed between the "project ABC" and "drawing" folders although both of them represent the subject of a document's content. Therefore, it is necessary to identify the possible semantics of a document so as to construct an ontology to interpret it in the Semantic Web environment.

In this research, the types of document were identified from the perspectives of how they were used in a construction project:

(1) From the Construction Project Stages in Which Documents are Used

The types of project documents can be classified according to the construction stages in which the documents are used. Generally, the lifecycle of a project can be broken down into several stages. In this research, the Process Protocol model was adopted because:

1) It provided a generic project process framework. Individual projects can create their own project process map based on the Process Protocol framework.

2) This construction process model was validated by being used in several projects, such as the Manchester airport building.

The Process Protocol model broke down a construction process into four stages, ten phases and ten activity zones (sub-processes). The stages included *pre-project, pre-construction, construction and post-construction/completion.* The phases included *demonstrating the need, conception of need, outline feasibility, substantive feasibility study & outline financial authority, outline conceptual design, full conceptual design, production design, procurement & full financial authority, production information, construction, and operation & maintenance.* Table 3-1 lists high level project documents used in each process phases. Activity zones included *Development Management, Project Management, Resource Management, Design Management, Production Management, Facilities Management, Health & Safety, Statutory and Legal Management, Process Management, Change Management.* Activity zones generally overlapped and were interactive. For example, Design Management often had important input in the Production Management and Facilities Management activity zones. The definition of each stage, phase, and activity zone is given in the Process Protocol project guide (Process Protocol, 2002).

Information evolves throughout construction project lifecycle in versions of documents. The phases and activity zones was used as a clue to identify information of a document in this research.

Table 3-1: Common Process and Documents of a Construction Project (Process Protocol, 2002)

Phase	Process Phases	Doc	Documents Used
0	DEMONSTRATING THE NEED	1	Statement Of Need-Initial
		2	Outline Business Case-Initial
		3	Risk Register-Initial
		4	Risk Management Process Plan-Initial
		5	Stakeholder List-Initial
		6	Communication Strategy-Initial
1	CONCEPTION OF NEED	1	Statement Of Need-Finalised
		2	Stakeholder List-Updated
		3	Outline Business Case-Updated
		4	Communication Strategy-Updated
		5	Project Brief-Initial
		6	Site And Environmental Issues Report-Initial
		7	Feasibility Design Brief
		8	Risk Register-Updated
		9	Risk Management Process Plan-Updated
2	OUTLINE FEASIBILITY	1	Outline Feasibility Studies
		2	Outline Business Case-Updated
		3	Communication Strategy-Revised
		4	Risk Management Process Plan
		5	Project Brief-Updated
		6	Site And Environmental Issues Report-Updated
		7	Risk Register-Initial
3	SUBSTANTIVE FEASIBILITY STUDY & OUTLINE FINANCIAL AUTHORITY	1	Outline Business Case-Updated
		2	Communication Strategy-Updated
		3	Procurement Plan-Initial
		4	Concept Design Brief
		5	Risk Register-Initial
		6	CDM Assessment-Initial
		7	Risk Management Process Plan-Revised
		8	Site And Environmental Issues Report-Updated
		9	Substantive Feasibility Studies
		10	Project Brief-Revised
4	OUTLINE CONCEPTUAL DESIGN	1	Full Business Case-Revised
		2	Project Brief-Revised
		3	Cost Plan-Initial
		4	Communication Strategy-Revised
		5	Outline Concept Design
		6	CDM Assessment-Revised
		7	Procurement Plan-Updated
		8	Risk Management Process Plan-Updated
		9	Risk Register-Revised
		10	Site And Environmental Issues Report-Updated
5	FULL CONCEPTUAL DESIGN	1	Full Business Case-Updated
		2	Communication Strategy-Updated
		3	Project Brief-Updated
		4	Site And Environmental Issues Report-Updated
		5	Risk Management Process Plan-Updated
		6	Risk Register-Updated

		7	Full Concept Design
		8	CDM Assessment-Updated
		9	Maintenance Plan-Initial
		10	Cost Plan-Updated
		11	Procurement Plan-Updated
6	PRODUCTION DESIGN, PROCUREMENT & FULL FINANCIAL AUTHORITY	1	Business Case-Finalised
		2	Communication Strategy-Updated
		3	Project Brief-Updated
		4	CDM Assessment-Updated
		5	Site And Environmental Issues Report-Updated
		6	Production Design
		7	Procurement Plan-Updated
		8	Cost Plan-Updated
		9	Maintenance Plan-Updated
		10	Risk Management Process Plan-Updated
		11	Risk Register-Updated
7	PRODUCTION INFORMATION	1	Communication Strategy-Finalised
		2	Project Brief-Finalised
		3	Procurement Plan-Finalised
		4	Site And Environmental Issues Report-Finalised
		5	Production Information
		6	'Construction-Ready' Site
		7	Maintenance Plan-Updated
		8	Handover Plan-Initial
		9	Risk Register-Revised
		10	Cost Plan-Finalised
		11	H&S Plan-Finalised
		12	Risk Management Process Plan-Revised
8	CONSTRUCTION	1	As-Built Information
		2	Risk Register-Finalised
		3	Completed Facility
		4	Handover Plan-Implemented
		5	Maintenance Plan-Operational
		6	H&S Plan-Operational
		7	Risk Management Process Plan-Revised
9	OPERATION MAINTENANCE	1	Maintenance Plan-Updated
		2	Risk Management Process Plan-Finalised
		3	Post Project Review
		4	Facilities Lifecycle Review

(2) From the Producers and Users of Documents

The types of project documents can also be identified according to the project partners by whom the documents are produced or used. Project partners assume different roles (horizontal fragmentation) throughout the entire construction process (vertical fragmentation) (Howard et al, 1989). Exchanges of information between

major players such as project managers, architects, contractors, engineers occur very frequently, in the forms of letters, orders, drawings, etc. Due to fragmentation, each party has its own information repository (i.e. information systems and database), exchanging information at the cost of doubled manual input (as shown in Figure 3-2) (Emmerik, 2000).

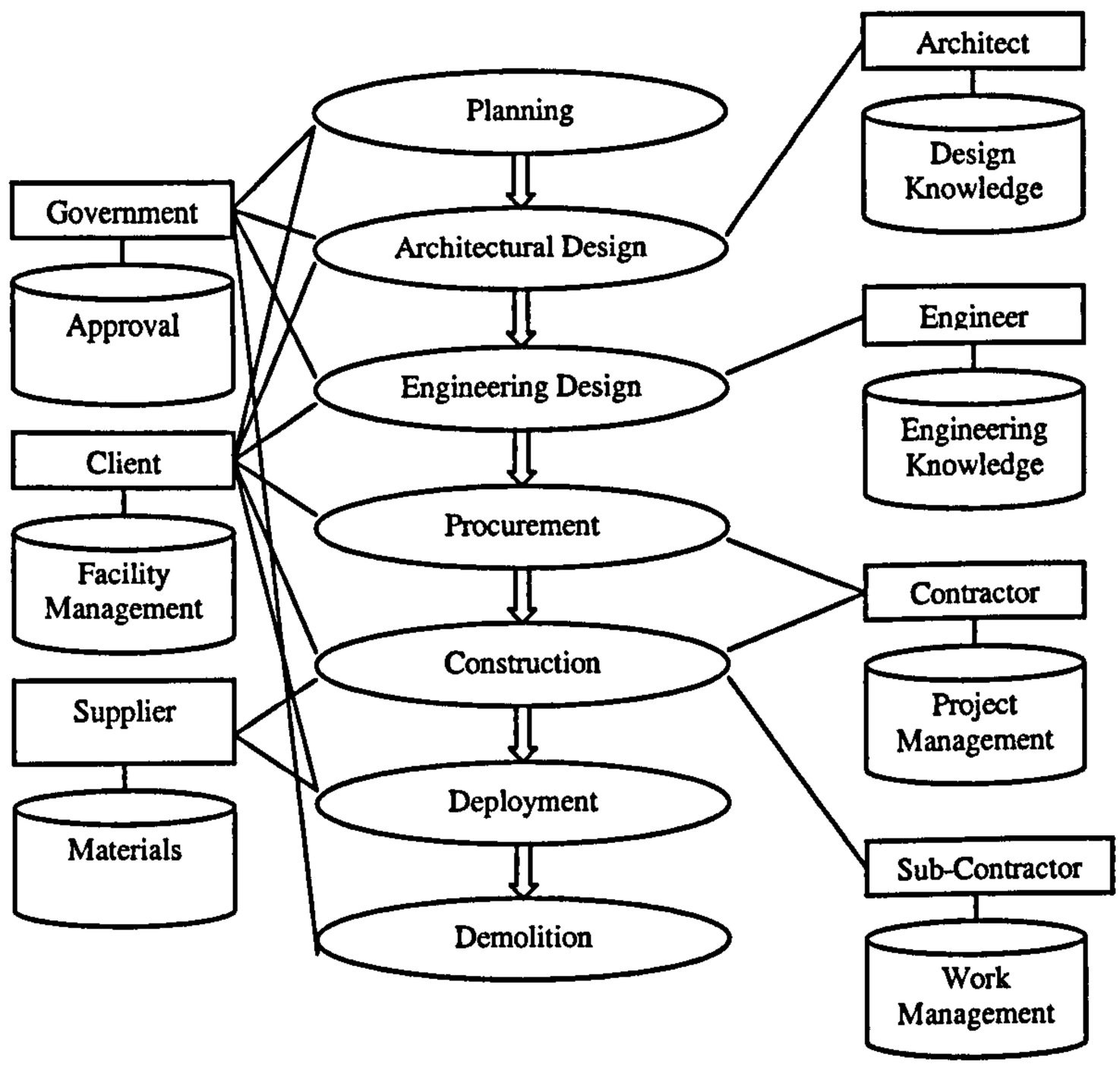


Figure 3-2: Fragmentation during Project Phases and among Partners (Emmerik, 2000)

Meanwhile, some shared documents overlap parts in two or more information systems/databases (as illustrated in Figure 3-3). For example, both architects and structural engineers use British Standards with different interests. Structural engineers keep a copy of architectural drawings as references. Project managers are interested in both design drawings and cost estimation. The arrows in Figure 3-3 illustrate the relationships between project partners and documents. Their relationships are useful in the document distribution/retrieval and were used in the annotation of documents in this research.

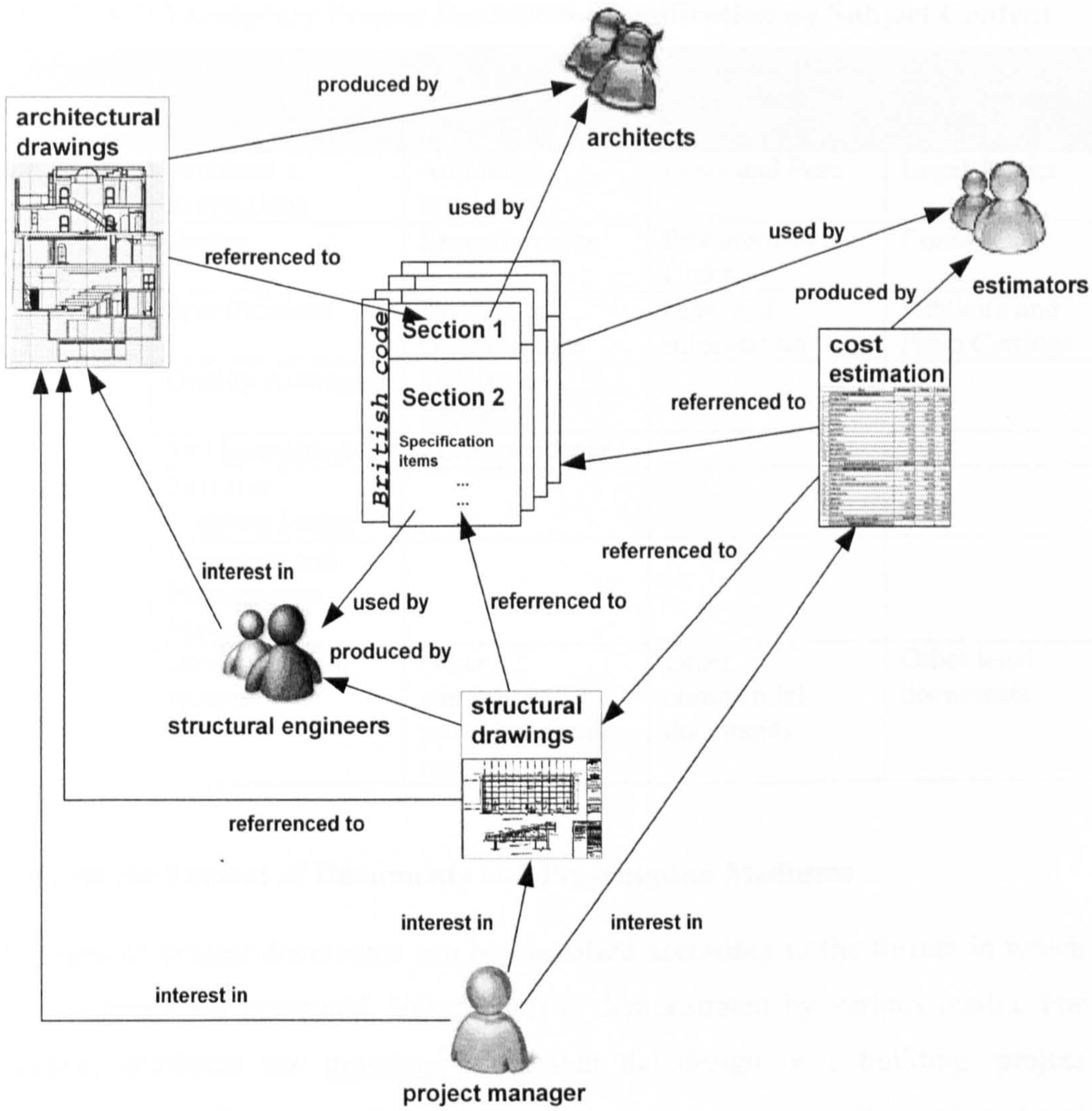


Figure 3-3: Project Documents and Project Partners

(3) From the Subject of Documents Content

The types of project documents can be classified according to the subject that the documents contained. From this perspective, documents could be classified into four major categories: *technical documents*, *project administration documents*, *commercial documents* and *legal documents*. In Table 3-2, some typical project documents are classified. In this research, these four subjects were used in annotating the semantics of construction project documents.

Table 3-2: Exemplary Project Document Classification by Subject Content

Document Type	Technical Documents	Project Administration Documents	Commercial Documents	Legal Documents
Sub-types	Architect's Instructions	Archiving Instructions	Costs and Fees	Legal Advice
	Design	Execution Plan	Procurement Order	Contracts
	Specifications	Partners and Organisations	Financial Information	Publicity and Press Cuttings
	Quality Assurance	Milestone Reports		
	Site Instructions	Correspondence		
	Test and Commissioning			
	Operation and Maintenance Manuals			
	Other technical records	Other administration procedures and records	Other commercial documents	Other legal documents

(4) From the Format of Documents and Presentation Mediums

The types of project documents can be classified according to the format in which the documents are presented. Information is demonstrated by various media. For example, architects use drawings to present the design of a building, project managers use videos to capture the event on a construction site, and material suppliers use text and pictures to demonstrate their products. Drawings, videos, text and pictures are mediums to present document content. Meanwhile alternative formats are also available to present information. For instance, textual information can be recorded into *.txt, *.doc, or *.pdf format. Typical mediums and formats are listed in Table 3-3. The document formats are not numerable due to the diversity of applications and software. Managing information in electronic documents, its format and data structural should be explicit to computers so as to be opened and edited. Computers then can accurately parse the content. Generally, information presented in the same medium has common properties, while information presented in the same format is processed by the same software. Therefore, annotating documents with associated medium and format enables the automatic processing of information in the Semantic Web environment.

Table 3-3: General Electronic Project Document Presentation Medium and Format

Presentation Medium	Format
Text	*.TXT, *.pdf, *.doc, *.RDF,*.RTF, TeX
Presentation	*.ppt, *.swf
WEB	*.html,*.htm, *.XML, *.VRML
Calculation	*.xls (Spreadsheets)
Database	*.Mdb (tables)
Drawing	*.drw, *.dwf, *.RDF
Audio	*.mp3, *.rm
Video	*.avi, *.wav, *.mpeg
Others	Special documents generated by other professional software

(5) Discussion

Major document types are reviewed in Section 3.6.1. The review covers the necessary information for humans and computers to process a document, such as the subject content it belongs to, the process stage it is applied to, the format and medium it is recorded in, and the project partners it is interested by. However these are not sufficient to represent the semantics of documents. Many links between documents are not represented, such as the structural elements to which the documents relate, the information source from which the documents are proposed, and the succeeding documents which a document leads to, also play important role in specifying documents' attributes/properties. These semantics of construction project information can be represented in the ontologies which annotate document's attributes/properties in the Semantic Web environment. The terms/taxonomies used to create the ontologies can refer to construction information models and classification systems. The major existing models are reviewed in the following section. IFC was adopted as the taxonomy to develop ontologies (in Chapter 6) for this research. These ontologies were used in the annotation of project document (in Section 5.4) so as to manage the project information semantically (as demonstrated in Chapter 7).

3.6.2 Construction Product Information Modelling Mechanism

There are two projects that have put efforts into the standardisation of construction product modelling: STEP (Standard for the Exchange of Product model data) by the International Standards Organisation (ISO), and the IFC (Industry Foundation

Classes) by the International Alliance for Interoperability (IAI). Applying the standards and XML (eXtensible Markup Language), several projects proposed schemas to classify and define construction product information: bcXML from the LexiCon (the development of a taxonomy for building and construction industry) by the European project eConstruct; the aecXML (Architectural, Engineering and Construction XML) initiated by Bentley Corp, USA; and the ifcXML (Industry Foundation Classes XML) developed by the IAI (International Alliance for Interoperability).

(1) Standard for the Exchange of Product Model Data (STEP)

ISO 10303 STEP, which stands for STandard for the Exchange of Product model data, was a comprehensive ISO standard, developed by ISO committee TC184/SC4 (ISO, 1992). STEP is a collection of Application Protocols (AP) which intended to create an international standard for computer-based description and exchange of the physical and functional characteristics of digital products throughout their lifecycle (Santos and Hernandez-Rodriguez, 2000). It is a set of standards for methodologies to develop product models. It was initiated by the manufacturing industry and adopted by the construction industry. The EXPRESS definition language (ISO 10303-21:1994) (ISO, 1994 a) and the STEP physical file format (ISO 10303-21:1994) (ISO, 1994 b) are two well-known standards widely used in product modelling projects. The Building Construction Core Model (BCCM) (Part 106), Building Element Using Explicit Shape Representation (AP 225), Building Structural Frame-Steelworks (AP 230); and Building Services-Heating, Vent, and Air Condition (AP228) are several parts of STEP being developed for the construction industry (ISO, 2005).

(2) Industry Foundation Classes

IFC, which stands for Industry Foundation Classes, was delivered by IAI in parallel with STEP. IAI is an international organisation of over 600 companies and software vendors in the construction industry. IFC proposed universal codes for modelling building elements, which were shared by the different kinds of software applications throughout the lifecycle of buildings. The IFC specifications have been developed following the STEP-based implementation methods, especially the EXPRESS definition language and the STEP physical file format.

The IFC model uses a strict referencing hierarchy. There are four conceptual layers (Figure 3-4). The first and lowest layer is Resource Classes (8-side shapes), which are used by classes in the higher levels. The second layer provides a Core project model, containing the Kernel (triangle) and several Core Extensions (squares). The third layer is the Interoperability layer (oblong shapes), which provides a set of modules defining common concepts or objects across multiple applications or construction industry domains. These three layers together define the platform layer. The fourth and the highest layer is the Domain/Applications Layer (rounds), which provides a set of modules tailored for specific construction industry domain or applications. This is also called extensible layer, because the schemas on this layer are extensible and new schemas can be defined on top of the platform for applications. Figure 3-4 illustrates the latest version of IFC architecture-IFC2x Edition 3 (IAI, 2006 b).

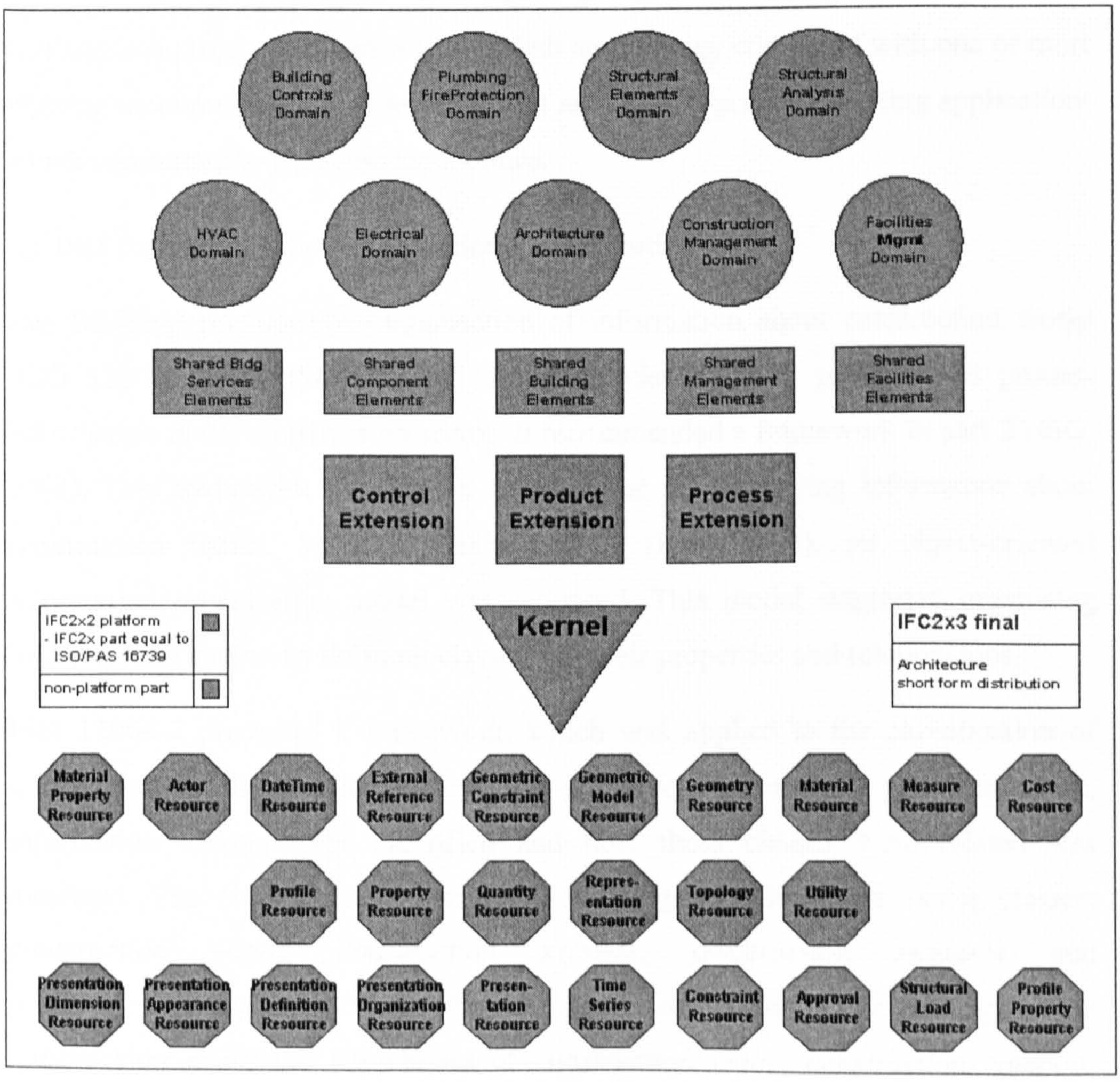


Figure 3-4: IFC2x3 Architecture (IAI, 2006 b)

STEP is a generic mechanism for product information modelling and IFC focuses on the construction products modelling. Most construction project information models (Classification Systems and Taxonomies) were developed based on them. Four major models are reviewed in the following section.

3.6.3 Construction Project Information Classification Systems and Taxonomies

Several research efforts have capitalised on establishing construction classification systems and information models. Their models were used in construction information taxonomies, which is useful in developing ontologies. This section reviews four representative information models/taxonomies/classification systems in the construction/building domain. They are ISO 12006 model, LexiCon Taxonomy, IFC model and UniClass (Unified Classification for the Construction Industry). SWIMS (Semantic Web-based Information Management System for construction projects) intended to establish an ontology compliant with one or more existing taxonomies so as to interact with as many as possible existing applications which committed to the same taxonomies.

(1) ISO Framework for Information Classification

The Building construction-Organisation of information about construction works (ISO 12006) is an ISO standard aimed to classify both product and process information in the construction sector. It recommended a framework in part 2 (ISO, 2001). This framework identified a set of tables for classifying information about construction works. In ISO 12006 part 3 (ISO, 2004), an object-oriented information organisation model was proposed. This model suggested organising building information by defining classes and their properties and relationships.

ISO 12006-2 presented a framework, which was applied to the classification of information in the complete lifecycle of construction works. In this framework, information classes were identified and how these classes were related was indicated. The construction information is categorised into four major classes: *construction result*, *construction process*, *construction resource* and *property/characteristic*. These classes have sub-classes. For example, the *construction result* has sub-classes of *construction entity*, *construction complex*,

space, and *construction entity part*. Meanwhile, the *construction result* relates to some other classes, such as *element*, *design element*, *work result*, etc. The significant relationships between classes are illustrated in Figure 3-5. Please refer to ISO 12006-2:2001 for the definition, example and notes of each class.

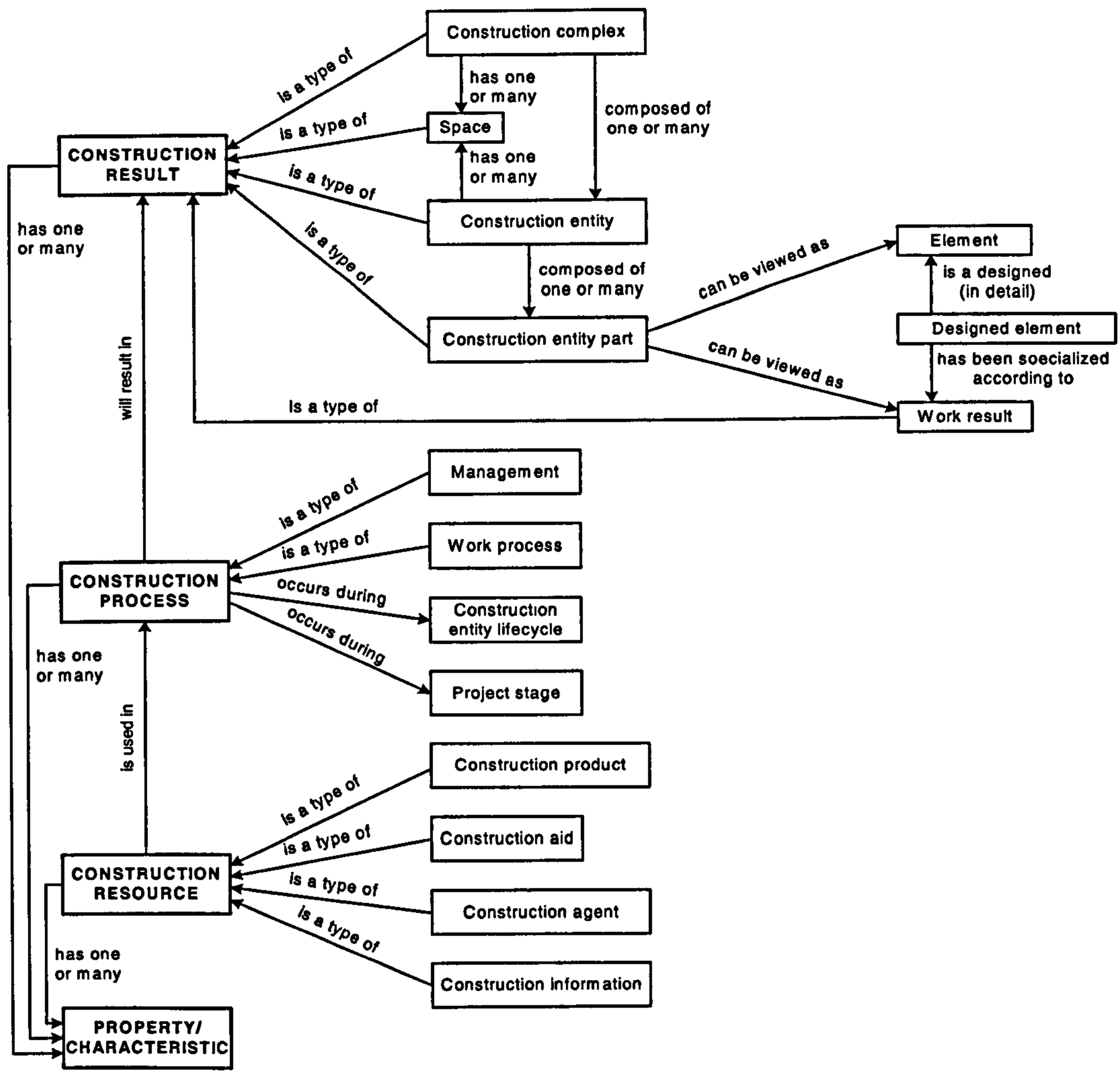


Figure 3-5: Classes and the General Relationships between Classes (ISO, 2001)

Figure 3-6 is a construction process model proposed in ISO 12006-2, which includes two dimensions of information classes, the Construction Entity Lifecycle Stage (vertical dimension), and Resources & Result (horizontal dimension). For example, in the *Production* Stage, Resources (*Construction products*, *Plant*, *Equipment*, *Operatives*, *Production managers*, *Production reference information*, *Production drawing*, *Specification*, etc.) are used to generate Result: *Produced construction entity*.

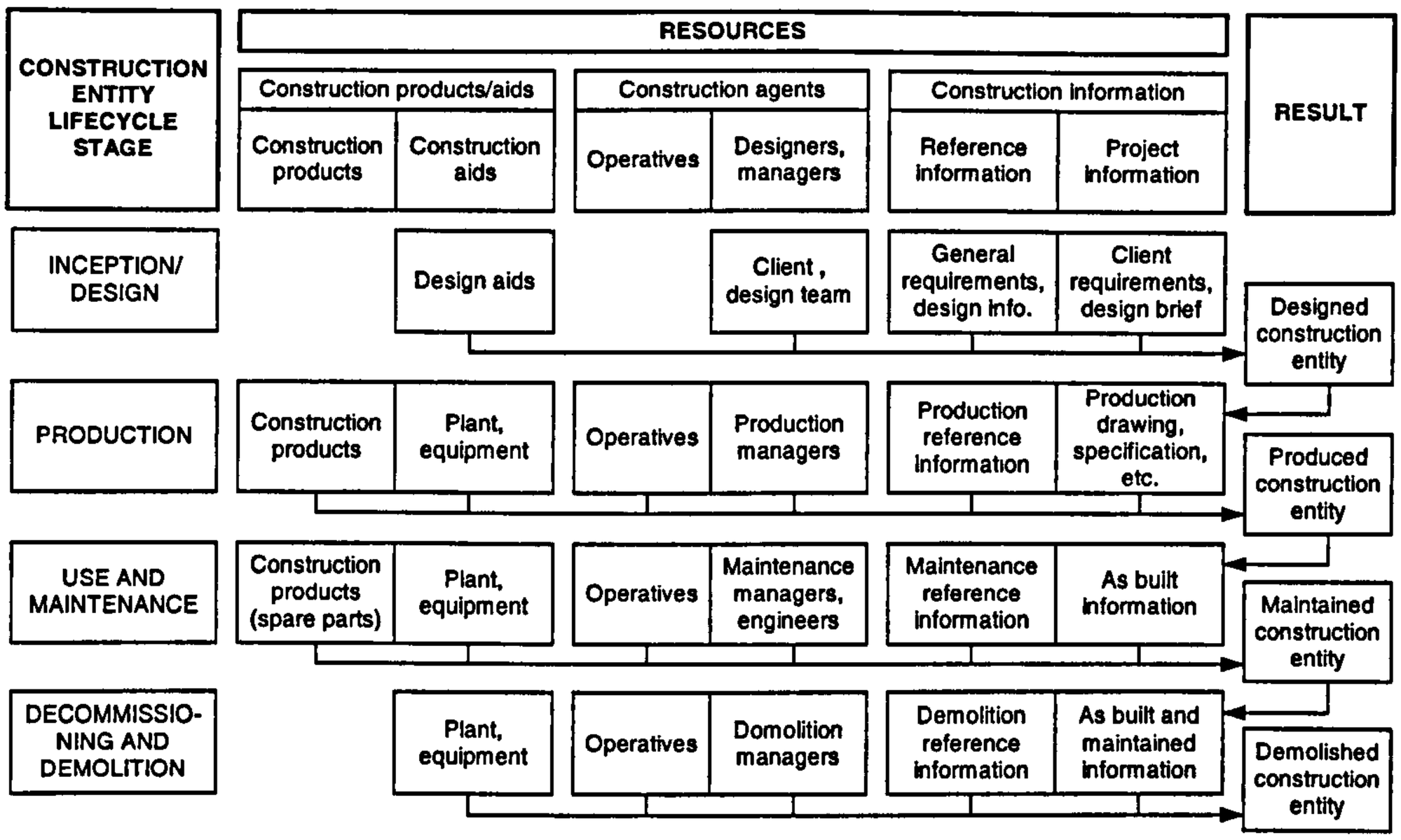


Figure 3-6: Construction Process Model (ISO, 2001)

Several classification tables were recommended for the development of the framework in ISO 12006-2. The principles of specialisation to each class were indicated in a table. For example, the principles to specialise Construction entity are *Form* and *Function or user activity*; principles to specialise Space are *Degree of enclosure* and *Function or user activity*; principle to specialise Construction agent is *Discipline*; principle to specialise Construction aid is *Function*; principle to specialise Construction information is *Type of medium* etc. Please refer to ISO 12006-2:2001 (ISO, 2001) for more about the classification tables.

ISO/DIS 12006-3 is complementary to ISO12006-2. The main part of it consists of the specification of a taxonomy model. In this model, a concept was defined in three ways: by means of properties, by grouping it to other concepts, or by defining its relationships with other concepts.

Entities of the model are derived from the Root concept and classified into three sub-concepts: objects, collections and relationships, which inherit properties from the Root (see Figure 3-7). The Root entity provides the ability to assign any set of names, labels, descriptions, references to its derived types, identifiers and dates. In this model, each object may have multiple names and this allowed for its expression in terms of synonyms or in multiple languages. Objects may be related to formal classification systems through the provision of references.

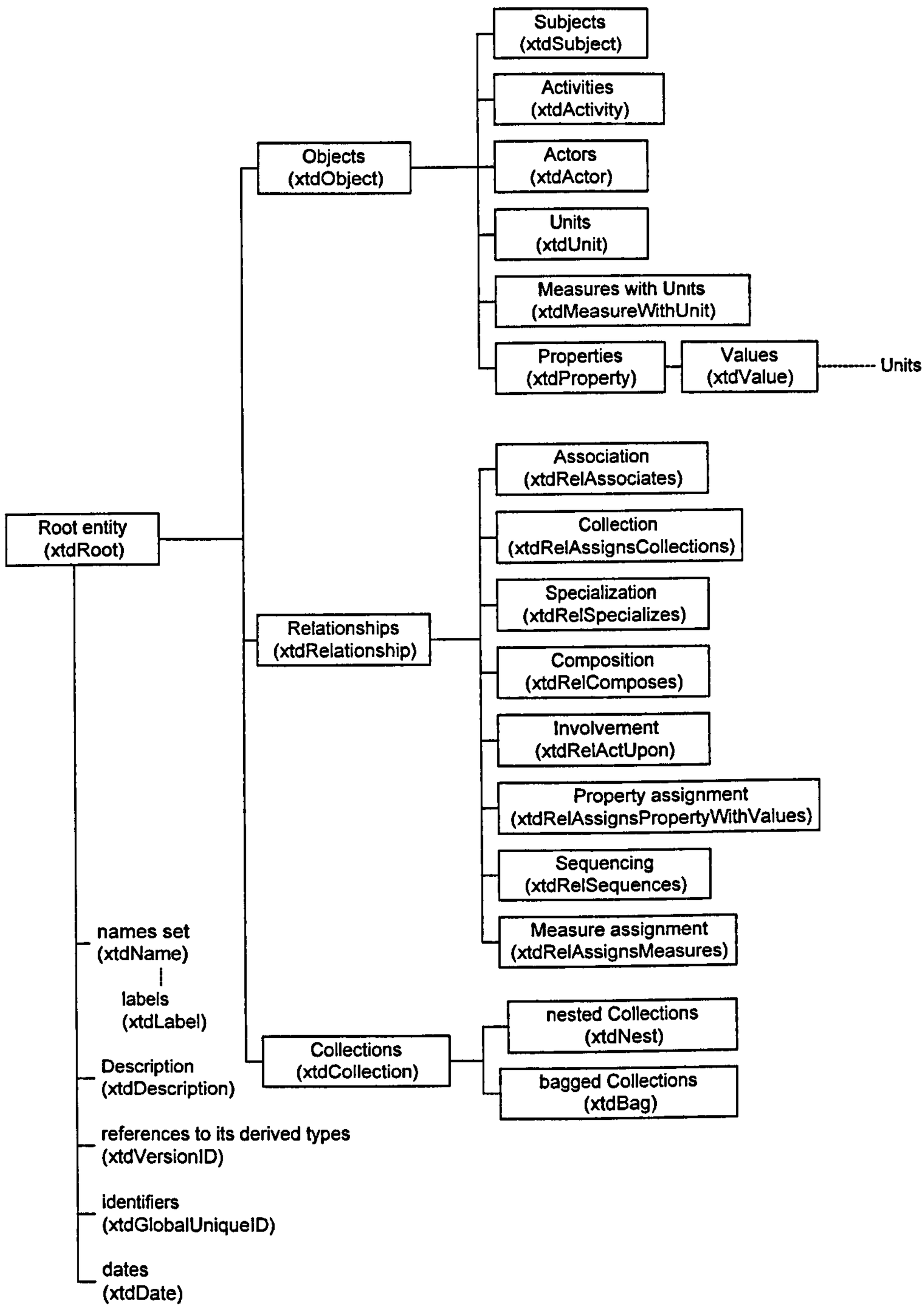


Figure 3-7: Entities of the Model in Standard ISO/DIS 12006-3

A set of properties, associated with an object, provided the formal definition of the object and its typical behaviour (the context for data). Properties are stored as values, which are optionally expressed in units. Properties are differentiated according to types of data containment: *Enumeration values*, *List values*, *Bounded list values*, *Bounded values*, *Single values* and *Table values*. The value content, associated to a Property through a Measure with Unit, will be stored in the Value component,

which is language dependent and therefore derived from the Language Representation class. The Language Representation class models the way any *Name*, *Description*, *Value* or *Reference* is represented on a per language base.

The model in standard ISO 12006-3 was described in diagrams conforming to the EXPRESS-G notation, and was described formally in the EXPRESS data definition language specification (ISO 10303-11:1994). Please refer to ISO/DIS 12006-3 for the specification of each concept.

(2) LexiCon and bcBuildingDefinition

The STABU Foundation has worked to develop a taxonomy for the building and construction industry called the "LexiCon", which was an implementation of ISO PAS 12006-3 and used by Europe project eConstruct. Entity classes in LexiCon can be illustrated in Figure 3-8. It classifies construction project objects into two categories: Concepts and Relationships.

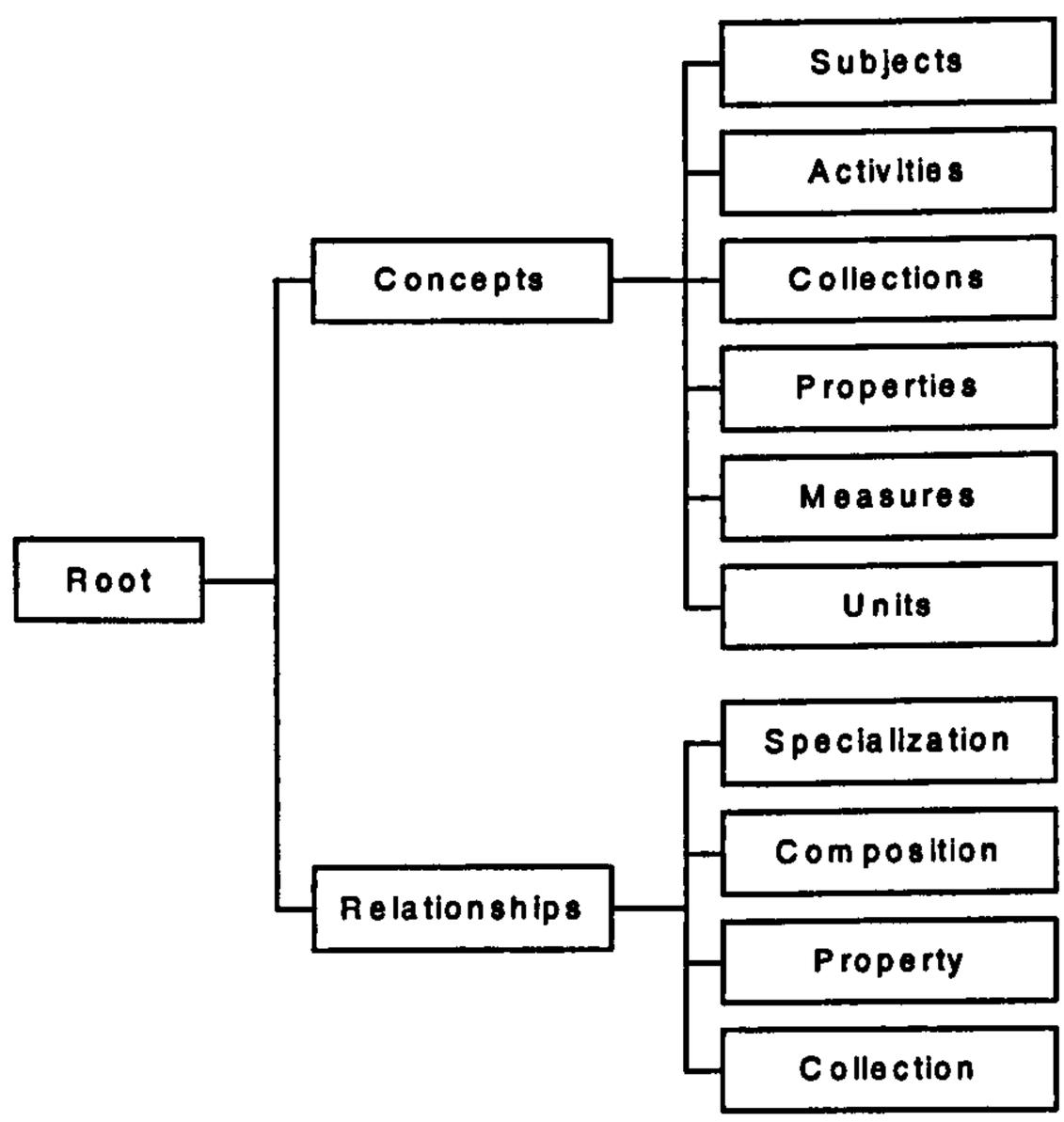


Figure 3-8: Objects Schema Breakdown by LexiCon

Six types of Concepts exist: Subject, Activities, Collections, Properties, Measures and Units.

Subjects: the objects that result from building Activities, including spaces and products, can be a physical thing or a logical thing.

Activities: the activities that result in or modify Subjects.

Collections: a grouping of Subjects, Activities or Properties based on a certain criterion, mostly a function or role.

Properties: characteristics or features characterizing or qualifying a Subject or an Activity, either directly or indirectly through a Property collection. A Property is a data container, the data being called the Value, and measured by a Unit.

Measures: the scale used for the value of a Property. Using specialisation of Units could replace the Measures.

Units: the scaling part of a Measure. This set of Units could at least be extended by one Unit, called class (note that the term class here is not the same as class in the meaning of type). Class Units are generally enumerations, defined in some document. Using a class as a Unit only makes sense with a reference to the document defining that class.

Above concepts can be represented in a concept hierarchy as shown in Figure 3-9.

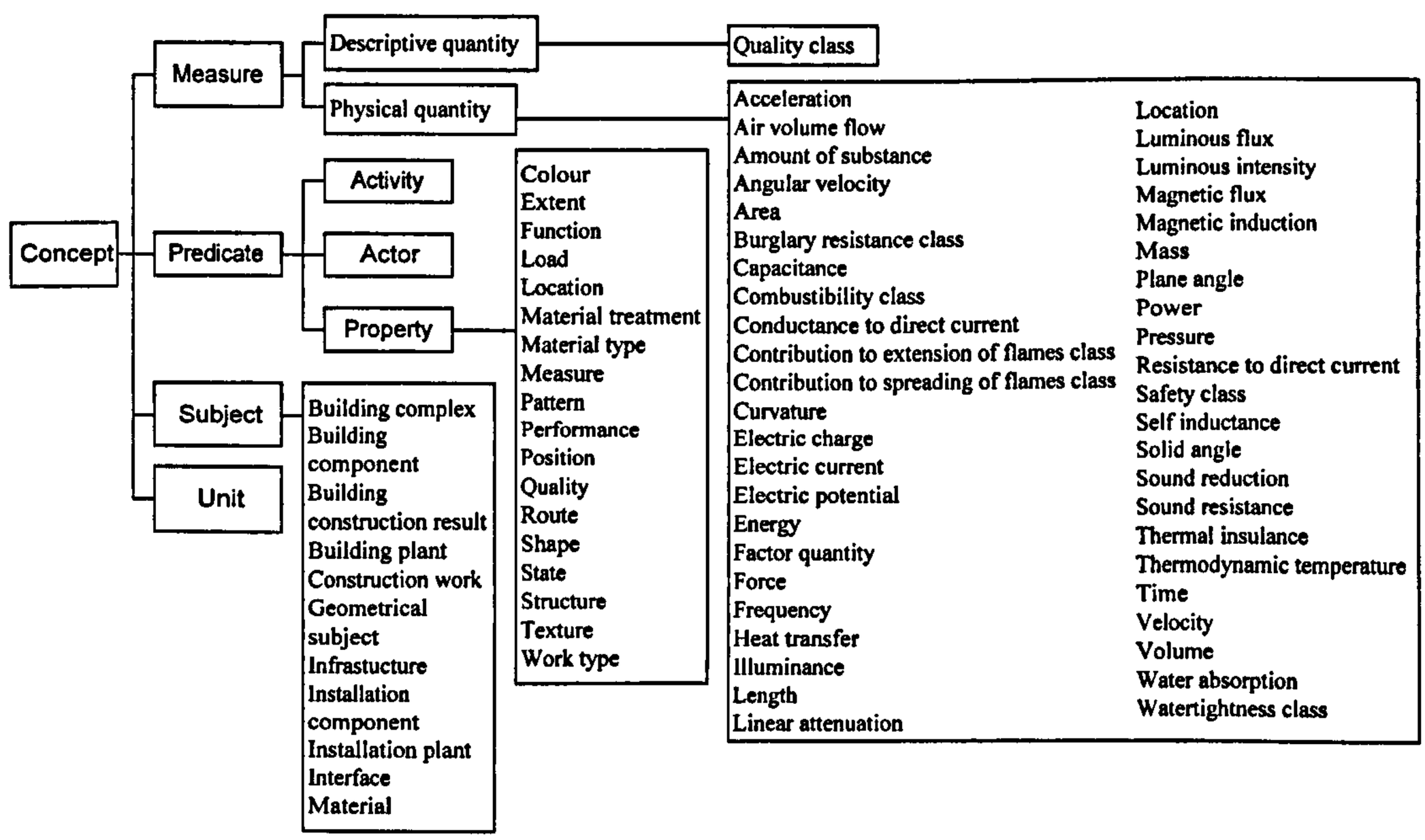


Figure 3-9: Object Hierarchy by the LexiCon Taxonomy Server (STABU, 2004)

Four types of Relationships included: Specialisation, Composition, Property and Collection.

Specialisation states that this concept is a specialised type of the other concept. The specialisation relationship is also called a 'type-of' relationship or an 'is-a'

relationship. Also the terms supertype and subtype are used, and in object orientation it is said that a subtype 'inherits' the characteristics of its supertype. Grouping classes according to specialisation results in a specialisation hierarchy.

Composition states that this concept actually consists of a number of components. This relationship is therefore also called a 'part-whole' relationship. Grouping classes according to composition results in a composition hierarchy.

Property states this concept can be characterised or qualified with a concept belonging to the category of Properties. The term 'Property' could be replaced by 'Characteristic' or 'Feature'. In the LexiCon Properties can be assigned to (associated with) Subjects or Activities, either directly or indirectly through a Property collection.

Collection states groups of concepts with the same type or category, hence there are collections of subjects, collections of activities and collections of properties, as far as these collections are neither specialisation nor composition relationships. In the LexiCon collections can be grouped into a collection hierarchy, which is a specialisation hierarchy.

A limited number of objects have been defined in the eConstruct project. They were called "bcBuildingDefinitions" taxonomy. Each member in the "bcBuildingDefinitions" taxonomy specialisation tree has a set of properties. A prototype application of the bcBuildingDefinition taxonomy in material procurement is available on the Web (bcXML, 2003). However, to complete the full LexiCon taxonomy, about 50 man-years should be consumed.

(3) IFC

Industry Foundation Classes (IFC) is a way of specifying "things" occurring in the built environment that can then be used by software authors to create the IFC compliant software applications. IFCs described the building model, its components, and the relationships between them in a single model. Therefore, a computer readable model, containing all the information of the building's parts and their relationships, is assembled and facilitates the information sharing between project partners throughout the lifecycle of buildings. The data elements of IFC Object Model represent the parts of buildings, or elements of the process. The relevant

information about those parts is also contained. This database will grow as the project goes through design, construction and operation. The latest release of IFC specification is IFC 2x Edition 3 (IFC2x3) (IAI, 2006 b) as introduced in previous sections (Figure 3-4). Its Kernel is the schema in the IFC model that establishes the information for all other sections. Concepts defined within the Kernel are in a hierarchical architecture. Derived from the Root, there are three fundamental entity types in the IFC model: Objects, Relationships, and Properties. They form the first level of specialisation within the IFC class hierarchy, as shown in Figure 3-10.

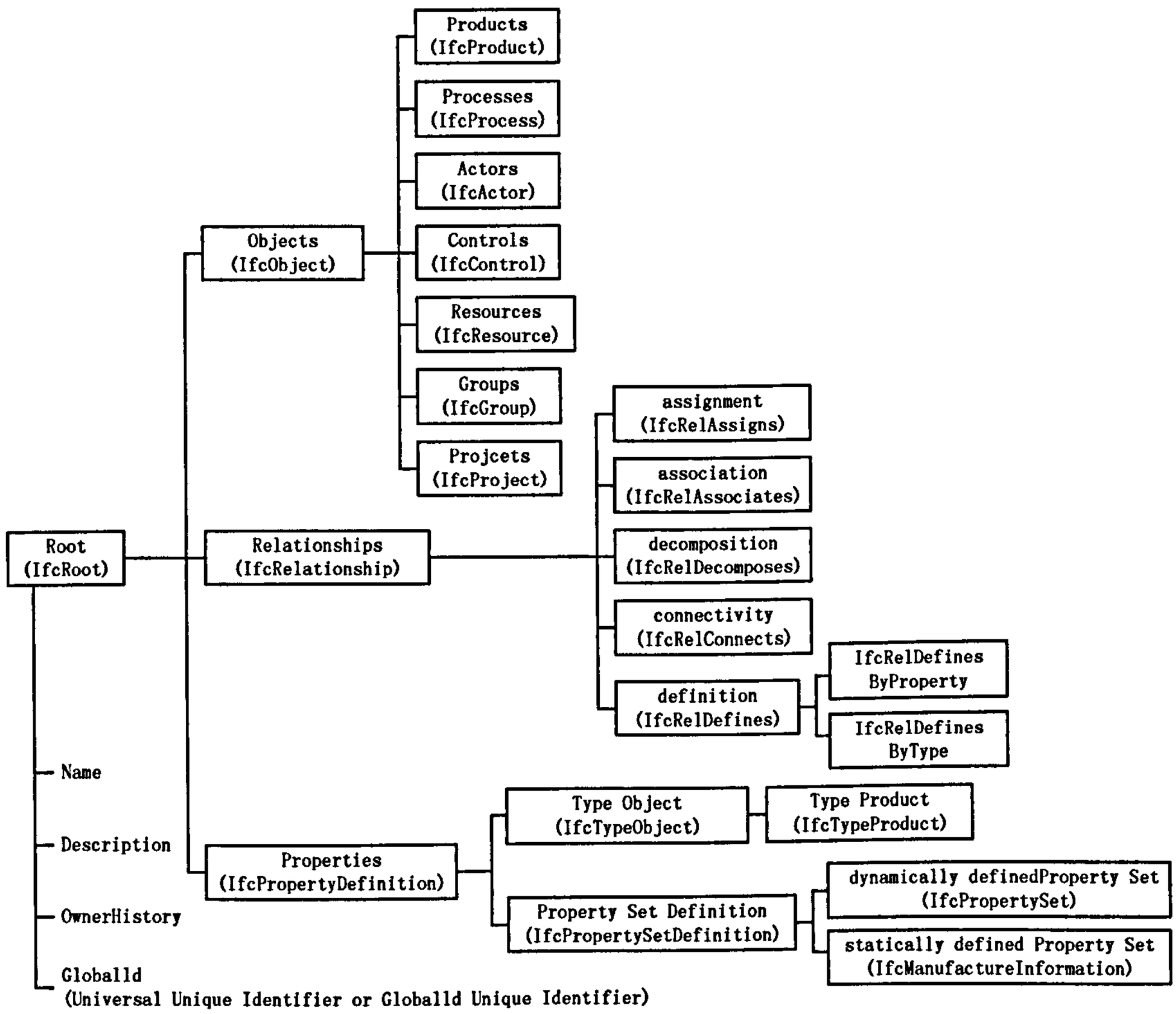


Figure 3-10: IFC Entity Schema Breakdown

Objects are 'things' that could occur in a constructed facility (including real things such as doors, walls, fans, etc. and abstract concepts such as space, organisation, process etc.). An Object in IFC stands for all physically tangible items (wall, beam or covering), physically existing items (spaces), or conceptual items (grids or virtual boundaries), processes (work tasks), controls (cost items), resources (labour

resource), or actors (persons involved in the design or construction process). There are seven entity types derived from Object. They are products, processes, controls, resources, actors, projects and group. IFC-based objects allow professionals to share a project model, yet allow each profession to define its own view of the objects contained in that model. IFC schema enables interoperability among professional software applications by using universal entity schema based on the IFC specification.

Relationships are the generalisation of all relationships among things or items that are treated as objectified relationships in the IFC model. An object gets its context information from the relationship in which it is involved. There are five relationship types in the IFC model derived from Relationship. They are Assignment, Association, Decomposition, Definition and Connectivity.

Properties are the generalisation of all characteristics (either types or partial type, i.e. property sets) that may be assigned to objects. Properties in IFC occur in three layers: core layer, interoperability layer and domain layer (in Figure 3-4). They reflect the specific information of an object type. The property definition is applied to the objects using the concept of Relationships. Three types of property definitions in the IFC derived from Property. They are type object and property set definition.

The type object defines the specific information about a type. It refers to the specific level of the well recognised generic - specific - occurrence modelling paradigm. The object style is represented by a list of property set definitions, where the order in the list implies a decreasing generality. The list of property sets describes the available specific information about the type object. Thereby the type object is used to define the common properties of a certain type (or style) of an object that may be applied to multiple instances of the same type.

The property set definition is a generalisation of property sets, which are either dynamically defined property set or statically defined property set.

The dynamically defined property set defines properties without entity definition of the properties existing within the IFC model. The declaration is done by assigning a significant string value to the name of the entity. This property set defines all dynamically extensible properties. It is a container class that holds properties within a property tree. These properties are interpreted according to their name attributes.

The statically defined property set defines properties for which an entity definition exists within the IFC model. The semantic meaning of each statically defined property is declared by its entity type and the meaning of the properties that is defined by the name of the explicit attribute.

Please refer to IFC 2X3 documentation (IAI, 2006 b) for more information.

(4) UniClass

UniClass (CPIC, 1997) is a classification scheme for the construction industry. It is intended for organising material library and for structuring product literature and project information. It is based on a framework developed by ISO technical committee TC59/SC13 WG2 in 1994, which is for an information classification system in the construction industry. UniClass classification system comprises 15 facets (i.e. information about building and civil engineering elements, spaces, documents, phases, and materials). Figure 3-11 demonstrates the 15 facets. Items can be shared among relevant facets. Applying UniClass, each facet can be used as a "stand alone" facet for the classification of a particular type of information. In addition, items from different facets can be combined to classify complex subjects.

To compare it with previous taxonomies/classification systems, Figure 3-11 depicts UniClass facets in a hierarchy. However, different from ISO 12006 model, LexiCon taxonomy and IFC model, which classify information into a hierarchical system, UniClass is structured with a faceted classification system rather than the hierarchical system, and partly use a hierarchical classification system within a facet to classify items in detail. The facets and items in the 15 UniClass can be roughly mapped to ISO 12006 and IFC. Facets A, B, and C are for general summaries concerning information form or management field. Facets D, E, F, G, H, and K consist of facilities, spaces, elements, and operations for civil and architectural works. Facets L, M, N, P, and Q are useful to classify information concerning construction products, materials, and attributes. The definitions of each facet are summarised in Table 3-4.

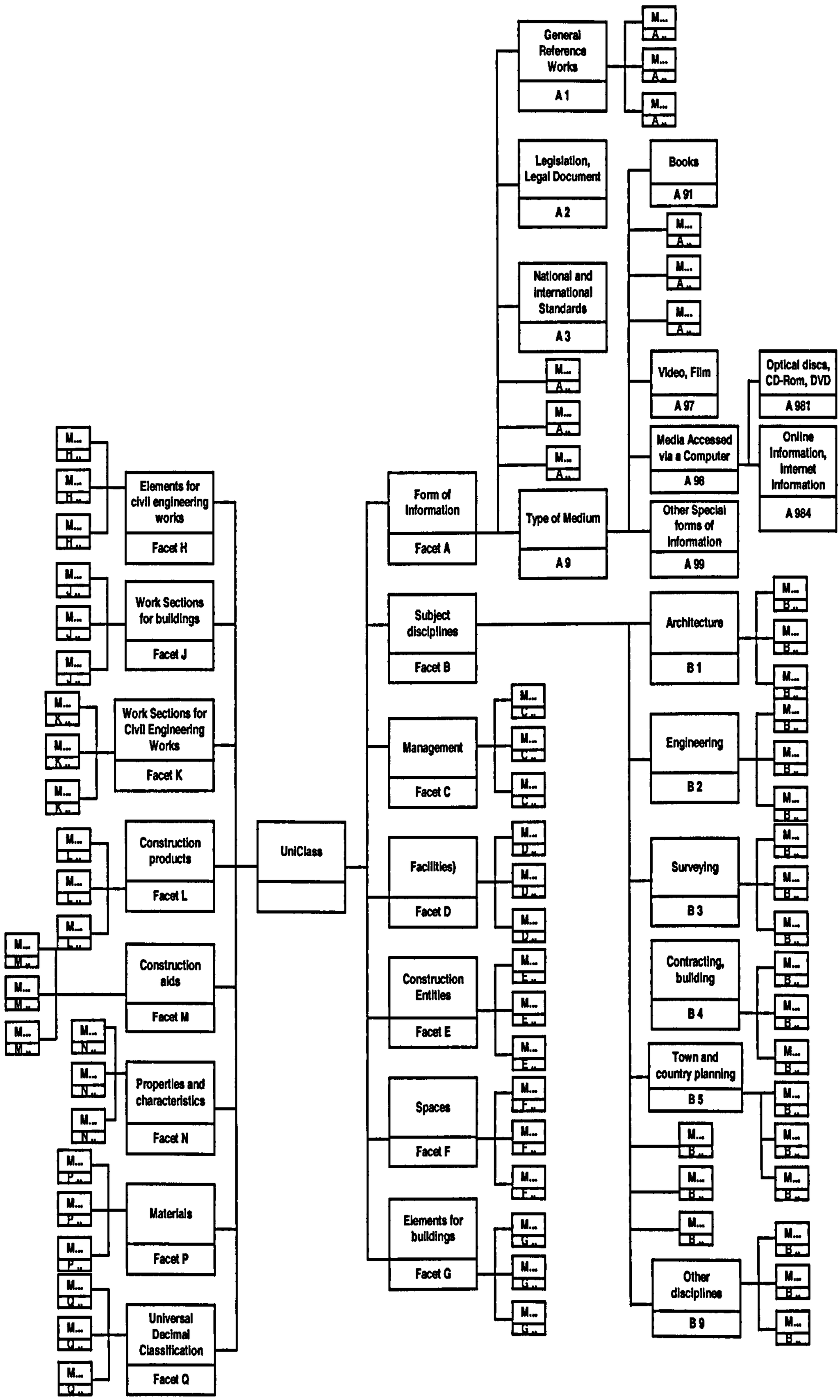


Figure 3-11: UniClass Table Lists

Table 3-4: Index of UniClass Tables (CPIC, 1997)

Facets	Description
A Form of information	This facet is useful for organising reference material in libraries, and also, when using combined codes, for denoting the medium in which information is published.
B Subject disciplines	This facet is useful where it is appropriate to organise information according to subject discipline.
C Management	This facet is intended for classifying project information according to the stage in the lifecycle of a project in which the information is generated.
D Facilities	This facet classifies construction works according to the user activity (or purpose) which they are intended to serve.
E Construction entities	This facet classifies construction entities according to physical form/basic function.
F Spaces	This facet classifies spaces according to a number of different characteristics including their location, scale, and degree of enclosure, but not according to user activity.
G Elements for buildings	This facet classifies major physical parts of buildings and can be used for organising both design and cost information.
H Elements for civil engineering works	This facet classifies major physical parts of civil engineering works; its primary use is for cost analysis.
J Work sections for buildings	This facet is based on CAWS and is used for organising information in specifications and bills of quantities, and for classifying information on particular types of construction operation.
K Work sections for civil engineering works	This facet is based on CESMM3 and has similar uses to those of Facet J.
L Construction products	This facet is used for classifying trade literature and design/technical information relating to construction products.
M Construction aids	This facet is for classifying trade literature and technical information relating to plant and equipment used for aiding construction operations.
N Properties and characteristics	This facet is used for classifying information on subjects related to properties and characteristics, for the arrangement of information in technical documents and as a qualifier to codes form other facets.
P Materials	This facet is for classifying different kinds of material, and also for adding as a qualifier to codes from other facets, especially Facet L.
Q Universal Decimal Classification (UDC)	This facet indicates how UDC can be used to classify subjects not covered elsewhere in the Uniclass system. Only the main headings of UDC are given.

UniClass is compatible with some other classification systems, such as the Electronic Product Information Co-operation (EPIC, which is used for product classification in electronic commercial system), the Civil Engineering Standard method of Measurement (CESMM, which is used for operations classification in quantity surveying for civil engineering project), and Common Arrangement of Work Sections (CAWS, which is used for architectural works in Europe).

3.6.4 XML-based Languages for Construction Project Information Schema

The **bcXML** was another outcome of the European project eConstruct. It aimed at developing the standards to support communication related to the procurement of materials, components, assemblies, documents, systems, services and equipments over national borders in Europe (bcXML, 2003). It provided the European building construction industry with a communication infrastructure in three aspects: supporting eCommerce between users and suppliers of building materials, components, systems and services; integrating with eCommerce and design/engineering applications; and supporting virtual market places over the borders of the individual European member states.

The **aecXML** was an initiative of Bentley Systems in 1999, then brought under the IAI North American Chapter. The purpose of aecXML was to enable communications between different software systems by establishing a standard way of structuring data for a construction project. By the time of its merge with IAI in 2001, it had only premature releases (IAI, 2002).

The **ifcXML** was issued by IAI to represent its IFC information model. Since January 1997 (IFC 1.0), several releases have been issued (IFC1.5 in 1997, IFC1.51 in 1998, IFC2.0 in 1999, IFC2.x in 2000, IFC2x2 in 2003, and IFC2x3 in 2006). The exchange of information via the STEP physical file format remains viable for all the areas of IFC (Liebich, 2001). The ifcXML is one of the widely used codes for software vendors in the construction sector.

3.6.5 Taxonomy in This Research

The terms used to name ontological elements (concepts, properties and relationships) constitute a taxonomy. A taxonomy and relations between elements constitute an

ontology. Ontology represents the real world information to computers. It is the backbone of the Semantic Web (Figure 2-1). Most construction project information is documented in diverse project files. Using the Semantic Web technologies, documented information is annotated. The annotation includes two parts: the characters of documents (by ontological concepts and properties) and the links between documents (by ontologies, the inherent relationships of concepts and properties). Therefore, a taxonomy model which represents building information in a structured schema is crucial in the annotation.

STEP suggested a standard series for exchanging digital data between product models. Parallel with it, IAI delivered IFC model for standardising the code of building elements between software vendors. IFC included a generic architecture for information-related domains and a building entity schema in ifcXML. Meanwhile, work groups of the ISO/TC59/SC13 (Technical committee of Building Construction/subcommittee of Organisation of information about construction works) produced two standards: ISO 12006-2, which defined frameworks for the classification of information about construction works, and ISO 12003-3, which specified a taxonomy model for building objects. In the Europe project eConstruction, taxonomy, LexiCon, for building and construction industry was developed based on ISO PAS 12006-3. The subset of LexiCon, bcBuildingDefinition taxonomy, encoded in bcXML, was defined for the material procurement application. ISO 12006 model, LexiCon taxonomy and IFC model used various rules to classify construction project information into a concept hierarchy. They are supposed to be compliant with each other although they are not yet totally compatible. All of them are under development and some studies on the co-ordination of these models are in process (Ekholm, 2005). Barresi et al (2005) produced a set of mappings amongst concepts from ISO 12006, bcXML and IFC Kernel model in FUNSIEC project (2006).

ISO 12006-2 was intended for use by organisations. It developed and published a classification model and some specialisation principle tables. It specified rough disciplines for the classification of information. Based on it, ISO/DIS 12006-3 was intended for the development of dictionaries used to store or provide information about construction works. It described a model in EXPRESS data definition language. The model included a framework for common terminology about objects

and attributes that were of interest for the construction industry. The common framework can be referenced by classification systems, information models, object models and process models.

LexiCon taxonomy defined building objects by concepts and relationships. Property is a sub-class of concept in this model. The application of it has been demonstrated in a material procurement case, which used "bcBuildingDefinition" taxonomy (the embryo of LexiCon) and bcXML. However, the full LexiCon taxonomy was not completed yet.

IFC model focused on standardised information expression for the development of software and databases. It was a major reference to bcBuildingDefinition. IFC provided a practical taxonomy for building information, including an Object hierarchy and Property sets. The ifcXML has been adopted by many projects, such as Tanyer and Aouad's project database (2005) and e-COGNOS ontology (El-Diraby et al, 2005).

Different from the above taxonomies/classification systems, UniClass provided a flat information classification system, which classified construction information into 15 facets. Items in UniClass facets are coded. UniClass facets and items can roughly be mapped to concepts in ISO 12006 and IFC models.

From four considerations, in this research, IFC model was adopted in the construction information ontology:

1) IFC is a vendor neutral standard for software interoperability between CAD systems and software developed by partners of project. Adopting IFC enables the developed system to interact with all IFC-committed applications and compatible with other IFC-based semantic resources, such as e-COGNOS ontology.

2) IFC entity schema most matches to the Semantic Web information annotation structure than the one in ISO/DIS 12006-3 and LexiCon projects. In the Semantic Web environment, information is annotated by RDF and ontology. RDF includes three parts: concept (asserted entity/object), properties of the entity and values of the property. The value could be numbers, strings or concepts. Meanwhile, concepts and properties are structured in an ontology.

That is the relationship between concepts and properties. ISO/DIS 12006-3 (in Figure 3-7) had three fundamental categories: *Objects*, *Collections* and *Relationships*. LexiCon (see Figures 3-8 and 3-9) had two fundamental categories: *Concepts* and *Relationships*. The *Properties* was a sub-category of either *Objects* in ISO/DIS 12006-3 or *Concepts* in LexiCon. In the IFC entity schema (Figure 3-10), there were three fundamental categories: *Objects*, *Relationships* and *Properties*. Here, as in the Semantic Web, *Properties* is a parallel entity with *Concepts*. The different position of *Properties* enables IFC-based building information models to easily participate in the Semantic Web environment.

3) The ifcXML is well defined and ready to use. ISO standards do not provide XML-based code and the bcXML only supports the material procurement domain so far.

4) The Collections and Collection entities in the ISO/DIS 12006-3 and LexiCon models could be easily confused. In the ISO/DIS 12006-3 model, Collections entity is parallel to the Object and Relationships. Its equivalent in LexiCon model is a sub-class of Concepts. The Collection entity is a sub-class of Relationships in both models. Collections is on behalf of a grouping of entities (subjects, activities or properties) while Collection states this relationship. Although Collections and Collection were defined as different concepts, it could cause confusion to users. In the IFC model, the equivalent concept to Collections is Groups (sub-class of Object), and the counterpart of Collection is Association (sub-class of Relationships). It was a minor change but the concepts became explicit.

The IFC model was adopted in this research because it is well developed, widely accepted, and compatible to the Semantic Web. IFC taxonomy was used in annotating information of project documents (and other document-related entities) and project partners (in Section 5.4). Terms from UniClass taxonomy, Process Protocol model and e-COGNOS ontology will enrich the IFC-base taxonomy to annotate project information. This taxonomy will be used in the ontologies developed for SWIMS (Semantic Web-based Information Management System for construction projects) (in Chapter 6). The adoption of standardised information taxonomy, IFC, would enable SWIMS ontology to be combined with other IFC-based

ontologies and interacting with all IFC-based applications, so as to integrate most of the existing databases, knowledge bases, software and applications into SWIMS. Eventually, the implementation of SWIMS will enhance the interoperability between independently developed software and knowledge bases.

3.7 SUMMARY

In this chapter, the concept and development of construction project information management were reviewed. Information management is a complex issue due to the fragmentation of the construction industry. Electronic documents became the major media for handling information between project partners in recent decades. The types of documents used in projects were then identified. Existing information taxonomies/classification systems were reviewed and discussed. Finally, IFC was adopted in this research because it is well developed, widely accepted, and compatible to the Semantic Web. IFC entity taxonomy was used in the ontology for annotating and processing information of project documents (with other document's attributes), project partners, and structural elements (in Chapter 6). Terms from UniClass taxonomy, Process Protocol model and e-COGNOS ontology will enrich the IFC-base taxonomy to annotate project information. The annotations will be used in the deployment of SWIMS (Semantic Web-based Information Management System for construction projects) (in Chapter 7). The adoption of standardised information taxonomy, IFC, would enable the combination of SWIMS ontology with other IFC-based ontologies and interacting with all IFC-based applications, so as to integrate most of the existing databases, knowledge bases, software and applications into SWIMS. Eventually, the implementation of SWIMS will enhance the interoperability between independently developed software and knowledge bases.

CHAPTER 4 - METHODOLOGY

4.1 INTRODUCTION

This project is an Information System (IS) research, which aims to investigate how to use a new technology for information management in a construction domain. This chapter explains how the research was carried out and justifies the various research methods chosen. It begins with a review of general IS research methodologies and then describes the "Building-Testing-Refinement" cycle, which was adopted for this research. The use of appropriate methodologies in eight research steps to accomplish the research objectives outlined in Chapter 1 is discussed. The identification of appropriate research methods and the justification for the adopted methods are also discussed, with the detailed use of each method being included in the relevant chapters.

4.2 METHODOLOGIES IN INFORMATION SYSTEMS RESEARCH

There are two major approaches to research methodology in social sciences: quantitative research and qualitative research. Quantitative methods are designed to provide summaries of data that support generalisations about the phenomenon under study (Moskal et al, 2002). Qualitative research aims to understand more about human perspectives and provides a detailed description of a given event or phenomenon (Creswell, 1997). Differences between quantitative and qualitative research are briefly described below (Bernard, 1995 and Leydens et al, 2004):

- Qualitative research seeks to better understand social or human problems that can be examined in their natural settings. Quantitative research uses few variables, many cases, and prescribed procedures to ensure validity and reliability. They often occur in a lab or other setting with precise, intentional controls and manipulations;
- Qualitative research features a holistic perspective. Quantitative research takes an analytic one, dividing the phenomenon into analysable parts;

- Qualitative research often shuttles iteratively between data collection and analysis using inductive methods. Quantitative research generally conducts analysis after data collection, working deductively;
- Qualitative research yields textual data (obtained from audiotapes, videotapes, and field notes). Quantitative research yields rich data that requires diverse data analysis techniques.

Each approach encompasses various research methodologies, which could overlap. The selection of which approach is appropriate in this research should be based upon the research question and available resources.

This research aims to investigate how to use the Semantic Web (a new information technology) for information management in a construction domain. It is an Information System (IS) research because in general, the main aim of conducting an IS research project is to "study the effective design, delivery, use and impact of information technology (IT) in organisations and society" (Keen, 1987). The "Building-Testing-Refinement" cycle was considered as a consensual term in classifying IS research methodologies (Galliers, 1992; Land, 1993; Shanks et al, 1994). A variety of research methodologies has been explored by researchers in the general field of IS. These are summarised in Figure 4-1, which is considered an appropriate framework for information systems research projects.

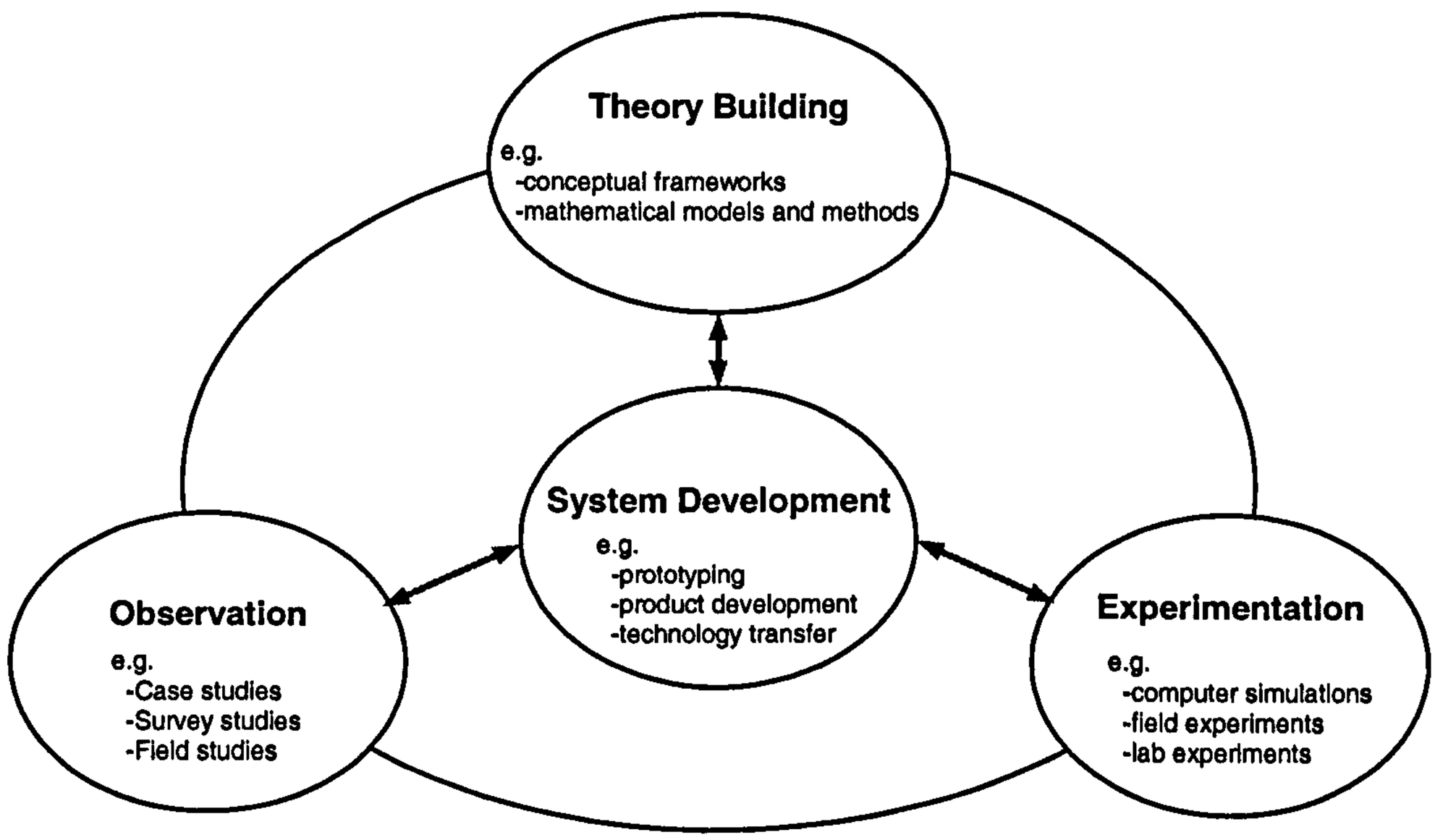


Figure 4-1: A Multi-methodological Approach to IS Research (Nunamaker and Chen, 1990-91)

Most of the methodologies shown in Figure 4-1 have been studied for a long time and Table 4-1 lists the definitions of the most common research approaches used in IS research. The methods adopted in this research were selected from this list.

Table 4-1: Definitions of Research Approaches Used in IS Research

Research Approaches	Definitions
Action research	A comparative research on the conditions and effects of various forms of social action and research leading to social action (Lewin, 1946).
Case study	An empirical inquiry based on observable evidence that investigates a phenomenon in a natural setting when the boundaries between the phenomenon and its context are not clear, using multiple sources of evidence (Yin, 1994).
Survey	A process for collecting information on only a sample from the population (Chambers and Skinner, 2003).
Laboratory and field experiment	They are experimental research methods. Laboratory experiments take place in laboratories or contrived settings, whereas filed experiments occur in real-life settings (Bryman and Bell, 2003).
Futures research	Any activity that improves understanding about the future consequences of present developments and choices (Amara and Salanik, 1972).
Simulation	An imitation of some real thing, state of affairs, or process. It uses one process to imitate another process (Hartmann, 1996).
Longitudinal study	A correlation research study that involves observations of the same items over long periods of time, often many decades (Wikipedia, 2006 c).
Phenomenological study	A descriptive investigation of lived experience that precedes attempts to provide theoretical explanations of the phenomena in question (Ilsley and Krasemann, 2002).
Ethnographic study	A method of observing people in their cultural context (Fetterman, 1998).

Above approaches are appropriate for different aspects of IS research depending on the domains described as follows:

4.2.1 Action Research

Action research (also known as participatory research, collaborative inquiries, emancipatory research, action learning and contextual action research) "aims to contribute both to the practical concerns of people in an immediate problematic situation and to the goals of social science by joint collaboration within a mutually

acceptable ethical framework" (Rapoport, 1970). The primary attribute of an action research is its focus on turning the people involved into researchers. The research takes place in real-world situations and aims to solve real problems. The application of action research methodology could combine various research methods, such as keeping a research journal, document collection and analysis, participant observation recordings, questionnaire surveys, structured and unstructured interviews, and case studies.

O'Brien (2001) indicated that although an action research is used in a real situation rather than in contrived and experimental research, it can be used for preliminary or pilot research, especially when the situation is too ambiguous to frame a precise research question. Lau (1997) reviewed the use of action research in information systems studies and proposed a term System Development (SD), which covers various methods used in analysis, design, development and implementation of information systems, and decision support systems. As a special type of action research, SD is deemed that the development of a method or system can provide "a perfectly acceptable piece of evidence (an artifact) in support of a 'proof'", where "proof is taken to be any convincing argument in support of a worthwhile hypothesis". SD could be thought of as a "proof-by-demonstration" (Nunamaker et al, 1990). SD research also been referred to as "engineering" type research (Cecez-Kecmanovic, 1994). However, it is still argued that if a SD methodology is a legitimate methodology in IS research (Burstein and Gregor, 1999).

Prototyping is also a type of action research method which is used in IS development (Baskervile and Wood-Harer, 1998). In the context of IS research, the theory/concept proposed usually leads to the development of a prototype system with the intention of illustrating the theoretical framework (Burstein and Gregor, 1999). In this sense, prototyping and SD are similar methods. The development of a concept/theory demonstrator or prototype is also a method of evaluation that is appropriate at the early stages of a software development life-cycle. It attempts to illustrate some or all of the proposed functionality of a system (Duke, 2001).

4.2.2 Case Studies

Case studies are done in a way that incorporates the views of the "actors" in the case under study (Zonabend, 1992). A case study explores complex real-life interactions

as a composite whole (Yin, 1994). The strengths of case studies lie in the fact that they allow for covering a large amount of ground for an acceptable cost (Fellows and Liu, 2003), and provide a means of looking in-depth at complex problems. Although a case study allows for the thorough examination of a particular situation, the results of such a study cannot be generalised beyond the single case (Yin, 1994). Another weakness of the case study approach is that the data collection and analysis process may be influenced by the researcher's interpretation of events, documents and interviews (Drake et al, 1998).

The case study approach has been used in numerous examples in various fields, such as Law and Medicine. In IS area, the case study method can be used to study IS failures or implementation efforts, study the impact of IT and IS on organisations, and research into the role and effects of IT and IS on society (Aziz, 2005).

4.2.3 Surveys

The survey method is used in empirical research. It is appropriate for analysis of groups or interactions; the collection of original data for describing a population too large to observe directly; investigating attitudes and orientations in a large population; and describing the characteristics of a large population. The design of a survey involves many interrelated decisions on such factors as identified by Kalton (1983):

(1) The mode of data collection, whether by interviews (telephone interview or face-to-face interview) or questionnaires (self-completion forms).

(2) The framing of the questions to be asked, whether by structured questionnaire (open or closed question, being compared in Table 4-2) or by interviews (structured or unstructured interviews).

(3) The method of processing the data whether it is quantitative or qualitative data (compared in Table 4-3).

(4) The sample design, whether random sampling, systematic sampling, stratified sampling, quota sampling, cluster sampling, multi-stage sampling, or probability sampling.

Table 4-2: Comparison between Open and Closed Questions (Foddy, 1994)

Open Questions	Closed Questions
1) Allow respondents to express themselves in their own words. 2) Do not suggest answers to indicate respondent's level of information; to indicate what is salient in the respondent's mind; and to indicate the strength of respondent's feelings. 3) Avoid format effects 4) Allow complex motivational influences and frames of reference to be identified. 5) Are a necessary prerequisite for the proper development of sets of response options for closed questions. 6) Aid in the interpretation of deviant responses to closed questions	1) Allow respondents to answer the same question so that answers can be meaningfully compared. 2) Produce less variable answers. 3) Present recognition, as opposed to a recall, for respondents and so respondents find them much easier to answer. 4) Produce answers that are much easier to computerise and analyse

Table 4-3: Distinctions between Quantitative and Qualitative Data (Saunders et al, 2003)

Quantitative Data	Qualitative Data
Based on meanings derived from numbers	Based on meanings expressed through words
Collection of results in numerical and standardised data	Collection of results in non-standardised data requiring classification into categories
Analysis conducted through the use of diagrams and statistics	Analysis conducted through the use of conceptualisation

The advantage of a survey lies in gathering data from a relatively large number of respondents within a limited time frame. It is thus concerned with a generalised result when data is abstracted from a particular sample or population (Naoum, 1998). The disadvantage is that little insight is usually obtained regarding the causes or the processes behind the phenomenon being studied. Also, Survey studies are subject to some well-known biases. For example, respondents may change their answers, either consciously or unconsciously, to show themselves in a better light or to conform to the expectations of those who are studying them. Taking the Likert scale questions (a type of psychometric response scale often used in questionnaires and is the most widely used scale in survey research) as another example, the respondent is asked to indicate their degree of agreement with the statement in a five-point scale. Respondents may avoid using extreme response categories (*central tendency bias*); agree with statements as presented (*acquiescence response bias*); or try to portray

themselves or their group in a more favourable light (*social desirability bias*). On the other hand, the experimenters can deliberately or inadvertently write survey questions that bias people to respond the way they want them to (Wikipedia, 2006 d). These biases should be brought to the attention of researchers in the survey design and result analysis.

In the IS research area, similar to the case study method, the survey method can be used for investigating the failures or implementation efforts of IS, the impact of IT and IS on organisations, and the role and effects of IT and IS on society (Aziz, 2005).

4.2.4 Laboratory and Field Experiments

The experimental research methods are best suited to known problems where the variables involved are identified or hypothesised with some confidence (Fellows and Liu, 2003). The key advantage of experimental research is the ability of the researcher to isolate and control a small number of variables that may be then studied in more depth. Its weakness is the limited extent to which the identified relationships exist in the real world. Laboratory and field experiments are used in an IS context for system experiments and implementation.

4.2.5 Futures Research

Futures research can develop intelligent forecasts concerning what is possible while indicating strategies for working toward desired goals. Technological forecasting is a subset of futures research. Technological forecasting includes "all efforts to project technological capabilities and to predict the invention and spread of technological innovations" (Ascher, 1979). Generally, a technological forecast includes four elements: the time of the forecast or the future date when the forecast is to be realised, the technology being forecast, the characteristics of the technology or the functional capabilities of the technology, and a statement about probability (Martino, 1983). Normally, futures research studies the impacts of IT and IS given different situations.

The Delphi, trend extrapolation, historical analogy, cross-impact analysis, simulations and models, environmental scanning and scenarios are seven major techniques used in futures research.

The **Delphi** method is designed for the systematic solicitation of expert opinion. It is characterised by anonymity, iteration with controlled feedback, and a statistical group response (Martino, 1983).

Trend extrapolation is based on empirical examination of a phenomenon with repeated measurements taken across time (Hill, 1978). A forecast can be generated by "observing a change through time in the character of something and projecting or extrapolating that change into the future" (Cornish, 1977).

The use of **historical analogy** in forecasting involves a "systematic comparison of the technology to be forecast with some earlier technology that is believed to have been similar in all or most important aspects" (Martino, 1983). Forecasting by analogy is one of the simpler and more common ways to forecast the growth of a new technology in social science.

Cross-impact analysis is more sophisticated than trend extrapolation. This method attempts to analyze one trend or event in the light of the occurrence or non-occurrence of a series of related events. A matrix is often used to facilitate this comparison (Welch and Watson, 1979). Cross-impact analysis enables the researcher to systematically examine the interactions among events, to organise the data descriptively, to use only a small number of input events and to test the outcomes against a variety of occurrences (Enzer, 1971).

A **simulation model** imitates one process by another process. The term "process" refers to some object or system whose state changes in time. This method is introduced in Section 4.2.7.

Environmental scanning is imperative for all types of environment libraries, as effective long-range and strategic planning require knowledge of anticipated trends and events.

Each of the above forecasting methods has its own advantages and disadvantages. Therefore, in many cases, it is helpful to combine several methods. Martino (1983) noted that **scenario construction** is an effective method for combining forecasts and forecasting methodologies into a holistic composite. Cornish (1977) described a scenario as: "it is simply a series of events that we imagine happening in the future." In other words, scenario writing is "making up stories about the future". As

explained by Schwarz et al (1982), the term "scenario" has numerous meanings. It can be used as a description for "a hypothetical, likely or unlikely, development or situation; a development which is described as caused to some extent by the actions and reactions of various actors: a desirable or non-desirable development or situation". A key assumption in scenario construction is that "there is no single best answer but a broad space of future possibilities" (Heijden, 1996).

In the IS research context, Galliers (1985) indicates that different impacts of IT and IS are identified by postulating different scenarios or futures. By accounting for a range of possibilities, scenarios can be distinguished from the other methods listed above. They do not generate or present the same degree of specificity, and have even been described as an "alternative to forecasting" (Schnaars, 1989).

4.2.6 Longitudinal Studies

Longitudinal studies repeated measures on same variables for the same group or groups on an extended series of occasions (Robson, 2002), and therefore the differences observed in those groups are less likely to be the result of differences across generations. In the IS research context, longitudinal studies can be used to identify the variables' changing relationships and their causes.

4.2.7 Simulations

As addressed under the section "futures research", a simulation model imitates and represents the system under study in the form of a set of mathematical variables and a number of explicit relationships between them. Simulations are closely related to dynamic models. The process is usually performed with the help of a computer. The computer simulation model can be a device for prediction, a method for deriving the future consequences of assumptions made about the present; a tool for learning how a system works, and a means of improving communication (McLean, 1978). Computer simulations also provide "a qualitatively new and different methodology" for the numerical experimentations (Rohrlich, 1991). In the IS research context, simulations can be used to see how a system works in the experimentation stage of a software/system development life-cycle.

4.2.8 Phenomenological Studies

Phenomenological studies aim to understand a particular phenomenon (Robson, 2002). It explores people's perception of an event rather than describing the event external to them. It provides a means of describing the inter-relationship of many factors found in real-life and recognises the fact that the researcher will interpret what is being studied in a particular way. The researcher's prejudice may affect the interpretation of reality and thus makes the research conclusions subjective (Galliers, 1985). This method can be used to study successful systems and system engineers' experiences.

4.2.9 Ethnographic Studies

Ethnographic research comes from the discipline of social and cultural anthropology where an ethnographer is required to spend a significant amount of time in the field. Ethnographers immerse themselves in the lives of the people they study (Lewis 1985) and seek to place the phenomena studied in their social and cultural context. Ethnography was discussed as a general approach to the wide range of possible studies relating to the investigation of information systems (Pettigrew, 1985). The use of ethnography in the study of IS includes information technology management (Davies and Nielsen, 1992), the development of information systems (Hughes et. al, 1992), and design and evaluation of information systems (Myers, 1999).

Major research methodologies have been briefly reviewed as above. The methods used in this research were selected from the above list, which "must be appropriate to the nature of the object studied and the purpose and expectations of the inquiry" (Sayer, 1992). In the following section, the research is designed based on in-depth understanding of the research question and objectives.

4.3 RESEARCH PROCESS AND METHODOLOGY ADOPTED

4.3.1 Understanding the Research Question

This research aims to investigate the applicability of a relatively new information technology to the construction domain. The research question is "How can construction project information management best be undertaken within the Semantic Web environment?" Answering this question requires an understanding of

the two objects, "the Semantic Web" and "Construction Information Management", their relationship and the underlying assumptions.

(1) Two objects

Two objects, "the Semantic Web" and "Construction Information Management" are addressed in this research. The definitions of the two objects were clarified by answering the following questions: "What is the Semantic Web?", "What is information?" "What is construction information?", "What is information management?", "How should construction information be managed?" and "Who manages construction information?"

(2) Relationship between the objects

The "how" question also indicates a relationship between the two objects. There is a *domain*-implementation-*technology* relationship between the two objects, as illustrated in Figure 4-2.

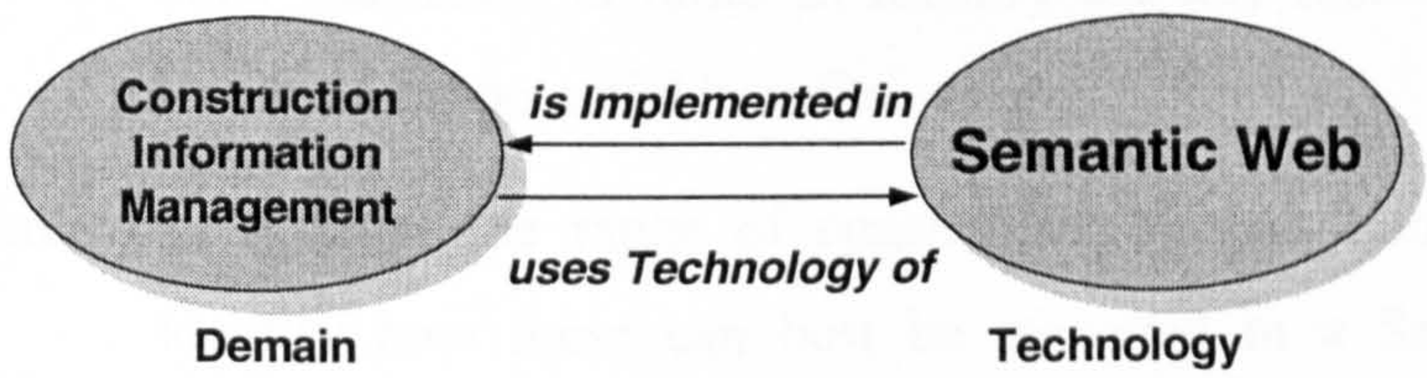


Figure 4-2: RDF of Research Question

The implementation process can be specified by answering the following questions: "What can the Semantic Web do?", "Which of these cannot be achieved by current construction information management methods?", "In what scenarios can the Semantic Web improve the efficiency of construction information management", "What software tools can be used?", "How should construction project information management be developed to take advantage of Semantic Web techniques?", "What new functions of the Semantic Web have not yet been explored by other researchers?", and "What new requirements does the application domain impose on Semantic Web techniques?"

(3) Assumptions

The research question implies an assumption that construction project information can be managed in a Semantic Web environment. It is rational because firstly, the Semantic Web is closely related to knowledge management, Web Services, Web 2.0,

and real-time enterprise and intelligent systems, which are current trends of IT that are being applied to the construction industry. This trend can be studied by employing historical analogy, which compares the Semantic Web with current World Wide Web technology. Secondly, this assumption can be justified by answering two questions: "What can the Semantic Web do?" and "Which of these cannot be achieved by current construction information management methods?" Based on the above assumption, this research focused on "how" to carry out the management activities in a Semantic Web environment and explored the applicability of Semantic Web techniques in the construction industry.

To answer the questions derived from the research aim, five research objectives were proposed, as specified in Chapter 1 and restated below:

Objective One: To review Semantic Web theories and background technologies.

Objective Two: To review implementations of the Semantic Web technologies in other industries in order to identify the key features, enablers, barriers, and potential benefits.

Objective Three: To explore the range of construction project information and identify how these can best be managed in a Semantic Web environment.

Objective Four: To implement a system framework and its middleware for managing construction project information in a Semantic Web environment.

Objective Five: To implement the prototype system using practical project information and evaluate the system demonstrator.

4.3.2 Research Steps and Methods

The research was designed following the "Building-Testing-Refinement" cycle. Figure 4-3 shows that the five research objectives were accomplished in eight major research steps. The research steps are listed in the middle of the figure while all research objectives are mapped to the corresponding major steps using white arrows. The colours/patterns filling in the step slots represent the main method(s) adopted in that step.

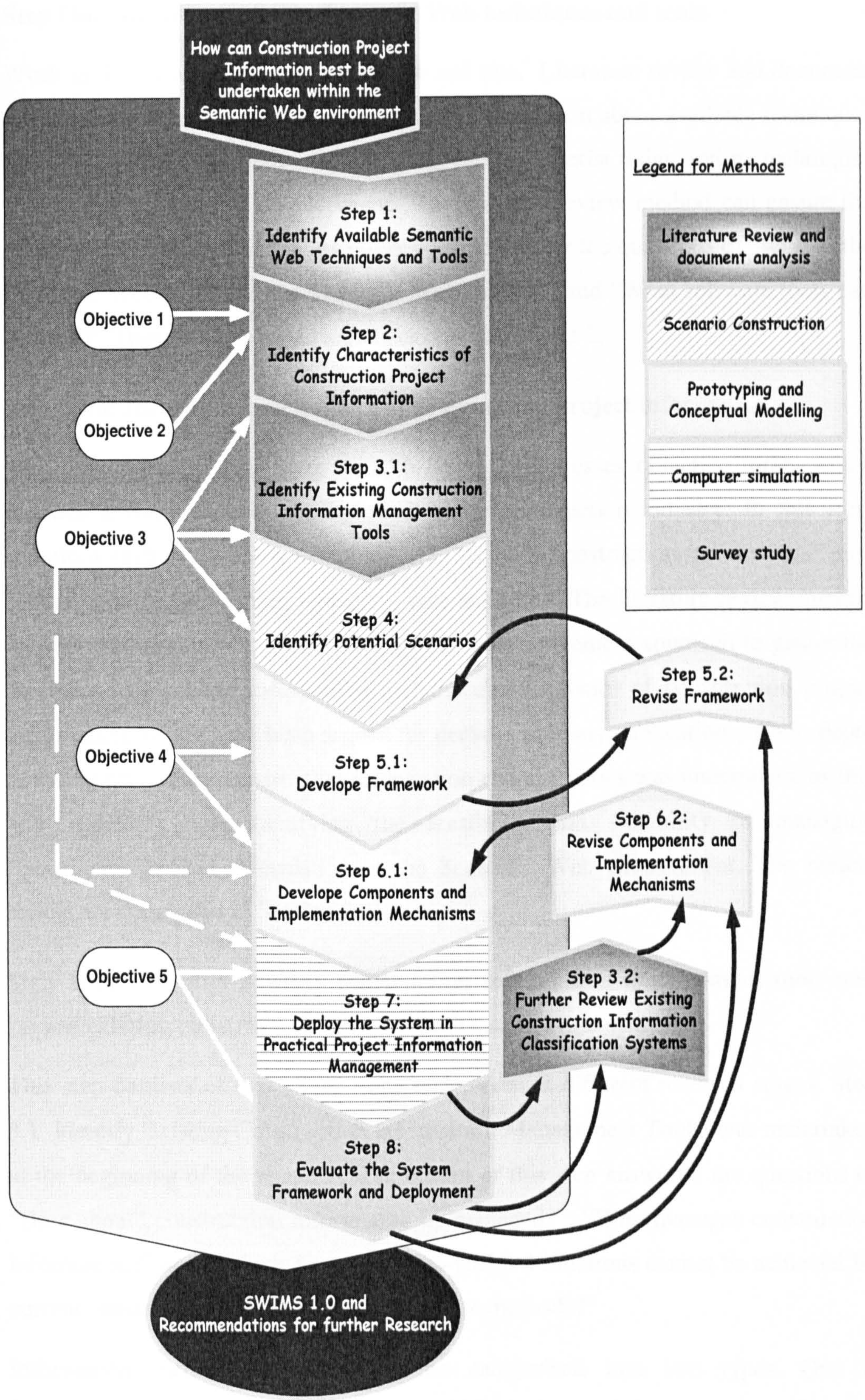


Figure 4-3: Research Steps and Methods Adopted

Step One: Identify available Semantic Web techniques and tools

Work in Step 1 addresses objectives one and two. Literature review and document analysis are major methods to collect related information about available techniques and tools because the Semantic Web does not really exist today and its techniques involved are under fast development. The literature review method can ensure the gathering of sufficient and latest information to answer the questions of "what is the Semantic Web?", "What the Semantic Web can do?" and "What software tools can be used?" These are explained and discussed in Chapter 2.

Step Two: Identify characteristics of construction project information

Step two, together with Steps 3, 4, 6.1 and 7, addresses objective three, which requires a comprehensive background in the construction industry. In this step, questions such as "What is information?", "What is construction information?" and "What is information management?" were answered. The literature review method was adopted in this step because it was the most convenient approach to gather the definitions of generic information and the characteristics of construction project information, which have been argued for decades in many publications. An in-depth literature review on construction information characteristics was undertaken, as this was necessary for identifying the feasibility and suitability of managing construction project information in the Semantic Web environment. The review results are concluded in Section 3.2.

Step Three: Identify existing construction information management tools and review existing construction information classification systems

This step consists of two minor steps undertaken at different research stages. Step 3.1, Identify Existing Construction Information Management Tools, was undertaken at the beginning of the research. The output of this step answered the questions of "How should construction information be managed?", "Who manages construction information?" and "Which Semantic Web-enabled functions cannot be achieved by current construction information management methods?"

Information management tools can be categorised into two types. One is information management techniques, which includes information management theories, models, methods, mechanisms, standardises, and information classification

systems. The other is IT tools adopted in the information management, which includes information management systems, applications, software and hardware. The knowledge about the first type of tools was gained from literature review because most of them are recorded in publications. The knowledge about the second type of tools was acquired by a literature review and the use of representative construction information management systems/software because the features of information management systems are not exact and detailed described in literature, while they can be gained through trial. Existing information management systems were reviewed in Section 3.3.

Step 3.2, Review Existing Construction Information Classification Systems, took place in two research stages. The first review, browsing through literature of existing information classification systems was carried out with Step 3.1. The second round review happened during the system development (ontology development in specific) stage when it is clear what exact knowledge was needed and could be reused in the Semantic Web-based system. This review focused on comparing available classification systems (refers to Sections 3.6.2 to 3.6.4). The selected classification system(s) was used in Steps 6.1 and 6.2. The main research methods adopted in this step were literature review and document analysis, complemented by scenario planning method and discussions with other scholars.

The potential benefits of the Semantic Web to the construction industry were clarified based on the knowledge of existing construction project information management tools and the Semantic Web. To achieve these benefits, the research focused on how to use the Semantic Web and how to structure construction information. It is a kind of technological forecasting research because the Semantic Web has not been established today. The possible applications of it in the construction industry should be identified first. The application scenarios were designed in the succeeding step.

Step Four: Identify potential scenarios

The scenario construction method was adopted to design and identify research scenarios because the feature of this research coincides with the assumption of the scenario construction method. The scenario construction method assumes "future is essentially unpredictable. Considering the uncertainties included in the future,

modelling will not lead into one future but rather to many different futures, each of which may be described in the form of a scenario" (Glenn, 2003). This research assumes the construction project information will be managed in a Semantic Web environment in the future. In the construction domain, diverse Semantic Web-based applications could be envisioned. The Semantic Web was too general to be thought as solving one key problem or creating one essential gizmo. As stated by Berners-Lee, "Will it enable new Killer Applications? No- it is intended to *be* a Killer" (Updegrove, 2005). Each application can be described in a scenario. In this research, six potential application areas (refers to Section 2.3.2) were reviewed and three scenarios (refer to Section 5.3) were designed, instantiated and evaluated.

The scenario design often combines with other research methods, such as case study and survey (interview and questionnaire), in order to identify all possible scenarios and key factors in each scenario. Because the Semantic Web is a new technology, which was formally proposed in 2001 (Berners-Lee et al), few people know about it and those that do are just aware of it in broad terms only, especially in the construction industry. The Semantic Web is envisioned as an intelligent Web which can access the meaning of information/knowledge and process it on behalf of human. This generic feature is just like a novel to most people. Also, the Semantic Web involves many techniques such as XML, RDF, Ontology, etc. These techniques are unfamiliar to project management experts as well. It is almost impossible to describe how the Semantic Web works on the project information before this concept is concreted into a prototype or demonstrator. Therefore, the construction practitioners, non-Semantic Web experts, could hardly give proper information to the scenario design. Then survey and interview methods are not appropriate at the current stage. A possible method to design the scenarios is to analyze related projects in other industries, reviewing their deliverables, journal papers, conference papers and project Homepages. The experiences from other industries could provide good references to the construction industry. The rationality of the scenarios was assessed via presenting the scenarios on academic conferences in the early system development stage and evaluating them with a prototype system demonstrator at the final stage of the research.

Once the application scenario had been determined, it was then necessary to develop an architecture for Semantic Web-based Information Management System (SWIMS)

for construction projects. This object (Object four) was accomplished in Steps five and six.

Step Five: Develop and revise framework

Information systems are major tools in information management. SWIMS is a natural intermediate linking the Semantic Web technology and real-life construction information.

As reviewed in Section 4.2, Information System research involves multi-methodological processes (Figure 4-1). Prototyping and conceptual modelling methods were adopted in the theory building process to establish the conceptual framework of SWIMS. The establishment of the framework and components models considered two factors: the functionality of the models and the appropriate techniques employed. The SWIMS framework is presented in Figure 5-2, which is similarly layered as the Semantic Web architecture (Figure 2-1). IDEF0 models were employed to define the system functions (refers to Chapter 7), identifying what exactly the developed system can do and how the information would be managed. The overall system framework consists of six components. Each component employed specific Semantic Web techniques.

The system framework was revised (Step 5.2) in two stages during the development of system components and implementation mechanisms, and after the system evaluation. The first revision enabled the framework to effectively support the system's functions. The second revision improved the overall system to be more practical for construction projects.

Step Six: Develop and revise components and implementation mechanisms

Once the overall system framework was determined, its components could be developed. In the system development process, the prototyping method and conceptual modelling method were adopted to establish a prototype system/demonstrator with the intention of illustrating the theoretical framework. Two key models were developed in this step, the information annotation model and the information ontology model. Properties of project information sources (documents and project partners) were identified and prototyped into annotation models. Associated information-processing mechanisms were defined. The

developed ontologies were constituted by project information taxonomy and relations of the taxonomy elements. A top-down development process was adopted in developing information taxonomy to ensure it complied with existing information classification systems. Eight ontology development methodologies were compared and Noy and McGuinness's methodology was adopted. This issue is explained in Section 6.2 in detail.

The annotation models, implementation mechanisms and ontologies were revised (Step 6.2) twice. One revision took place in applying the practical project information to the prototype system in a laboratory environment. This revision ensured the proposed models being of rationality and usability. The other revision took place after the evaluation stage. It improved the models from various aspects responding to the evaluators' feedback. As a result, the question of "How to develop construction project information for the use of Semantic Web techniques?" was answered and Objective four was accomplished.

Step Seven: Deploy the system in practical project information management

Step seven, together with step eight achieved Objective five. In the experimentation process, the prototype system was implemented on computers in the laboratory. The experimentation employed practical project information to simulate the process of system deployment in the proposed scenarios (refers to Chapter 7).

The computer simulation method was adopted in this stage because the objectives of the prototype implementation at the current stage were:

- To prove the proposed models and mechanisms and answer the question of "How to structure construction project information for the use of Semantic Web techniques?"
- To evaluate the proposed system at the early stage of the system development life cycle;
- To demonstrate the functions of SWIMS and answer the question of "What new functions the Semantic Web achieved which have not been explored by other researchers yet?"

A system demonstrator was generated in this step. The demonstrator captured the process of prototype system deployment, presented the main features of SWIMS, displayed the mechanisms employed in the information management, demonstrated the system functions, and showed the advantages of SWIMS over traditional information management systems. The demonstrator was used in the system evaluation.

Step Eight: Evaluate the system framework and deployment

The final objective was accomplished by evaluating two main research outputs: the system framework and the deployment demonstrator. The nature of the prototype decided the evaluation method adopted. Considering that the Semantic Web is a new technology, it takes time for evaluators to understand the technology and evaluate the system and SWIMS is at a very early stage, a Web-based multi-media evaluation method, which includes the advantages of other available system evaluation methods, was adopted. More details about the evaluation method are included in Section 8.3.1. Associated with the Web-based evaluation, an on-line survey method was adopted as the major approach collecting feedback from evaluators. A questionnaire was carefully designed, which followed a series of evaluation criteria (refers to Section 8.2). It used a mix of multiple-choice closed and open questions. Following the sampling logic of survey methodology, invited evaluators included major project partners and involved different stakeholders in the organisations (refers to Section 8.3.2). More importantly, considering the nature of the research, all invited evaluators had rich experience of IT and construction information management.

The responses were classified and analyzed upon corresponded evaluation criteria. The respondents' attitude to the system framework and deployment were gained via multiple-choice questions. The responses to open questions provided information about framework revision and models/mechanisms/ontologies revision. The feedback answered the question of "What new requirements to Semantic Web techniques are raised from the application?" Methodologies employed in the evaluation process are discussed further in Chapter 8.

Eventually, all five objectives were accomplished. The questions that arose from the research aim were answered.

4.3.3 Justification of Research Methods Adopted

The criteria of selecting appropriate research methods include:

- The method should be adequate to accomplish the research aim and objectives;
- The method should be reasonable at the current research stage;
- The method should be applicable based on available resources.

This research aims to investigate the applicability of the Semantic Web to the construction domain. To prove the applicability of the Semantic Web in construction information management, an information system is developed. Because the Semantic Web is a relatively new technology and under fast development, there is no mature system that can be directly transferred from other projects. The system developed in this research is the preliminary attempt for systematically managing the construction information in the Semantic Web environment. Therefore, this research does not exam specific variables and data of the information system. The research is a qualitative research and the methods adopted are applied to the early stage of the information system development.

The related knowledge of the Semantic Web is gained using the literature review method. This method is useful in gathering the latest information about the Semantic Web techniques and the sufficient knowledge to conduct succeeding research steps. Much knowledge about the construction information management method and mechanisms is recorded in publications and some of it has been standardised (such as the construction information taxonomies). A literature review is the most convenient approach to gather this knowledge. The literal information about existing electronic information management systems is limited because most of them are commercial software. The features of the software are generically described in publications and the operation details are protected against non-purchase acquisition. Also, the experience of using some specific software functions can only be gained through practise. The knowledge of existing information management systems was then gained using the literature review method and trying trials of representative software.

With all the background knowledge gained, an IS research was undertaken for managing the construction project information in the future Semantic Web

environment. Many methods are available to an IS research as depicted in Figure 4-1 and Table 4-1. The scenario construction method was adopted to identify the potential applications of the system because it is an effective method describing possibilities for the future than the other six alternative methods for futures research. The action research method, especially the prototyping method was adopted in developing the framework and semantic components of the system because it was more appropriate than other methods in establishing the fundamental components of a new information system in its early stage. The computer simulation method was adopted to operate the system and prototype ontologies in the laboratory environment because it can imitate the system implementation process with controllable variables, so that the research can focus on testing those Semantic Web-related functions and prevent the disturbing variables in the real world. In the system evaluation step, two popular evaluation methods, the case study method and the focus group method, were compared (as explained in Section 8.3.1) and considering the status of the developed system, the Web-based evaluation method was adopted which combines the advantages of the above two methods. An on-line survey method was adopted to collect evaluators' opinions as part of the Web-based evaluation. The case study, field experiment, longitudinal study, phenomenological study, and ethnographic study methods were not adopted because these methods often study information systems in real-life projects. However, before applying the developed system to the construction field, it was necessary to demonstrate the deployment of the system and prove its rationality, which is the major task of this research. A concept model and demonstrator can provide sufficient information for this purpose, while much further work is needed to apply the system to real-life projects, such as developing the end-user applications and Semantic Web techniques and tools, which requires a considerable amount of time.

Respecting the research aim and methods adopted, it is reasonable to assume that these methods are adequate to accomplish the research aim and objectives and are applicable at the current research stage based on the available construction information management and Semantic Web technologies and tools.

4.4 SUMMARY

The first part of this chapter presented available research methods for Information System research. The second part analyzed the implied question in the research aim. Employing the theory of "Building-Testing-Refinement" cycle, five research objectives and eight steps are proposed to answer the question and achieve the research aim. Considering the features of this research, the multi-methodological approach is adopted. The involved methods are presented in details in each research step. These include:

(1) Technology and literature review on the Semantic Web theory and construction information management tools;

(2) Document analysis on existing Semantic Web applications, construction information classification systems and practical construction project documents;

(3) Scenario construction for potential Semantic Web applications in construction information management;

(4) Prototyping the Semantic Web-based Information Management System for construction projects (SWIMS) and modelling construction information into ontologies;

(5) Simulating the operation of the prototype system on computers in the laboratory environment;

(6) Evaluating the system framework and deployment demonstrator by a Web-based multi-media evaluation approach. A survey method was adopted to collect evaluators' opinions.

Use of the selected methods and their role in the achievement of the research objectives is presented in subsequent chapters.

CHAPTER 5 - SWIMS FRAMEWORK, APPLICATION SCENARIOS AND ANNOTATIONS

5.1 INTRODUCTION

The construction project information management systems have great potential for improvement. To address problems in current information management in the construction sector, as highlighted in Section 3.4, a systematic perspective is needed. The Semantic Web-based Information Management System (SWIMS) is proposed to view heterogeneous construction information as a virtual body and understand the relationships and interactions between all the information entities, in particular project documents and partners. This chapter presents the framework of SWIMS and details some of the key components. The framework is intended to enable information management across diverse sources and parties. The relationships between the information are established and maintained by intelligent agents with well-defined ontologies, which facilitate both human and automated processing. The SWIMS implementation is described in the scenarios in Section 5.3. The crucial tasks in the deployment of SWIMS is identifying the semantics of information in construction management and developing the mechanisms for representing information by metafiles and ontology so as to use the Semantic Web tools. The annotation models for project documents and project partners are proposed in Section 5.4. The models concern fundamental semantics of documents and actors in the scenarios.

5.2 FRAMEWORK

5.2.1 Overview

Traditional Electronic Information Management Systems commonly consist of three components: Personnel, Data Repository, and Interface Application, as depicted in Figure 5-1. They support information management activities using computing software, which belongs to the Interface Application component, such as tracking construction processes in scheduling (e.g. Microsoft Project), cost control (e.g. Spreadsheets), design (e.g. AutoCAD) and communication (e.g. NetMeeting). As

argued before, these applications all have great potential for improvement in the Semantic Web environment.

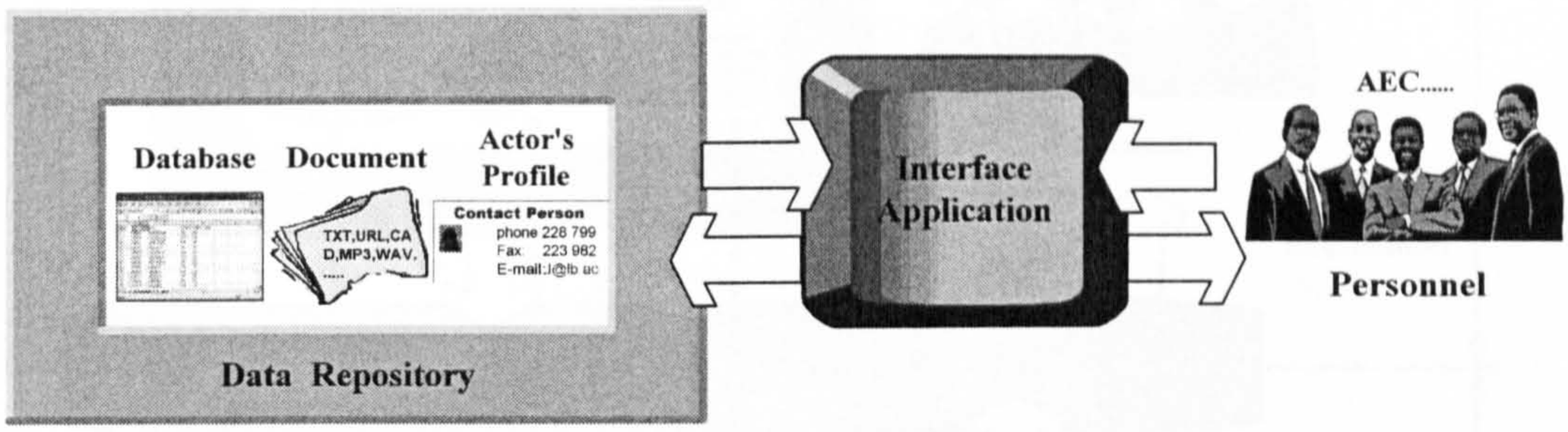


Figure 5-1: Framework of Traditional Information Management Systems

The Semantic Web has a layered architecture, with each layer representing a specific technique (illustrated in Figure 2-1). Implementing available techniques, the framework of SWIMS is similarly layered as depicted in Figure 5-2. This multi-layered architecture incorporates intelligent agents, ontology editors, rule reasoners, XML/RDF parsers, databases and knowledge bases, and software applications to facilitate the delivery of semantic-aware information/knowledge to project partners.

SWIMS consists of two major layers: the human interpretable information layer and the computer interpretable information layer. Each of these major layers contains several components.

(1) Human Interpretable Information Layer

This layer considers the information source issue of personnel, database, document and actor profile. Since people are one of the information sources and have the ability to create and manipulate project information, the Personnel component is the most dynamic element in the framework. Most changes in a project and requests for information are generated by this component. Databases, documents and actor profiles are major repositories of project information. They belong to the Data Repository component. The Interface Application component is the connection between humans and computers. It extracts and transfers information using software/applications (refer to Section 5.2.3). It could also be the existing information management tools refined to adapt to the Semantic Web environment. This component enables the progressive adoption of SWIMS in applications/software.

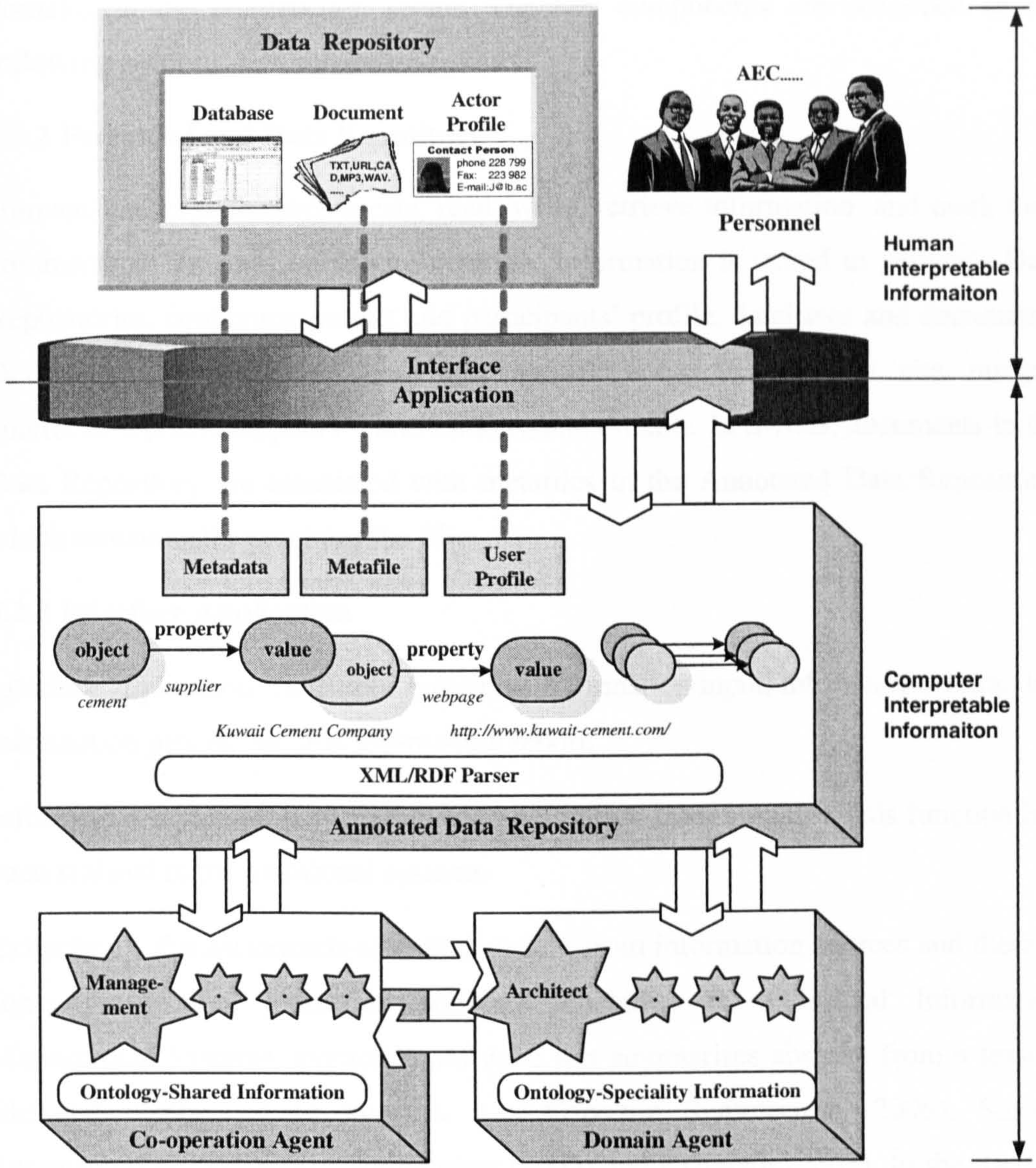


Figure 5-2: SWIMS Framework

(2) Computer Interpretable Information Layer

The Computer Interpretable Information Layer is the foundation of the framework and it includes three major components: Annotated Data Repository, Domain Agents and Co-operation Agents. The Annotated Data Repository contains a XML/RDF Parser, which annotates human interpretable information using the computer interpretable languages and then keeps the annotations in the metadata, metafiles and user profiles. The core of the framework lies in using intelligent agents to manage project information. The agents integrate ontology editors and rule reasoners handling ontologies for shared management issues or specific knowledge

domains in the construction sector. The key components are described in the following sections.

5.2.2 Personnel and Data Repository

Humans and organisations create, read, write, retrieve information, and mark their comments or interests on the information. Information is stored in partner's Data Repositories, containing project and participants' profile, databases and documents. Documents often include drawings, specifications, construction site images, contracts, reports, suppliers' certifications, and e-mails. In SWIMS, documents in the Data Repository are associated with metafiles in the Annotated Data Repository, which semantically annotates the files.

5.2.3 Interface Application

Interface applications fulfil four functions: information input, information extraction, information processing, and information report.

Information input software aggregates information from humans. This function has been realised in the traditional systems.

Extraction software extracts semantic content from information sources and there is some professional extraction software available. In traditional Information Management Systems, extraction software just summarises abstract from a textual file and displays it to humans, e.g. Copernic Summarizer (2006), Sinope Summarizer (2006), eXero Summariser (2005) and Extractor (2003). In the SWIMS, the extractor can extract semantics of files (textual, drawing or multi-media files) into metafiles and export it in human or/and computer interpretable formats for further use, such as the report generation. The Inxight SmartDiscovery Analysis Server (2006) is a fundamental semantic extraction software, which extracts entity, event and relationship from text and categorises it by predefined taxonomy. Inxight can export the analysis result in XML files. Some Web Services software have an integrated text extractor, RDF parser and ontology editor, e.g. Ontoshare (Davies et al, 2003). To date non-textual documents are annotated manually. Even if it takes time for engineers to annotate a document when no mature extraction software available, the metafile changes the way of processing information and saves time in

using information. Metafiles enable computers to access the content of documents for use in intelligence applications.

Intelligence applications focus on processing project information, which include Knowledge Management applications and Web Services applications (e.g. search, design, decision support, cost management, time management, to name a few). Comparing Figure 5-2 with Figure 5-1, although the interface of these applications may be similar to those in traditional information management systems, the backbone has been improved by the Semantic Web techniques.

In response to a query, an appropriate document list is reported to the user with the corresponding metafile, and then the content of documents is accessible without the need for humans to read the documents. Another instance to information report is report generator software, which automatically exports information into formatted documents. In traditional systems, it is humans who rewrite a report when information is changed. In SWIMS, computers will process it following given ontologies and rules. This saves human time and avoids misreporting when project information is frequently updated.

5.2.4 Annotated Data Repository

The Annotated Data Repository contains XML/RDF Parsers, which interprets the extracted semantics from information sources in RDF annotations and sends them to the ontology editors. The information sources include metadata from databases, metafiles from documents, user profiles from actor's profiles and metadata from RFIs (Requests For Information). Please refer to Section 2.2.3, Listings 5-1 and 5-2 for the RDF technique. Two annotation models are developed in Section 5.4 for the implementation scenarios presented in Section 5.3.

5.2.5 Intelligent Agents

Each project partner uses a personal agent and several shared agents. Each agent is supported by an ontology editor and a rule reasoner. Two types of agent: the Domain Agent and Co-operation Agent, exist according to their scopes.

A Domain Agent creates/collects speciality ontology of a specialty community (a group of individuals who complete a professional task, e.g. architects and structural engineers). Domain agents work on semantic mapping and consistency checking.

The semantic mapping adds concepts, relations, and instances to satisfy the purposes of the ontology while the consistency checking reconciles syntactic, logical, and semantic inconsistencies among the ontology elements. Consistency checking may also involve automatic classification that defines new concepts based on properties and constraints.

Co-operation Agents create/collect shared ontologies (e.g. commercial relationships and spatial relationships) of communities. These agents work on semantic mapping, consistency checking and logical reasoning, which find matches across various ontologies. As reviewed in Sections 2.4.2 and 2.4.3, many ontology editors and rule reasoners are available to achieve these functions.

5.3 SCENARIOS DEPLOYMENT

The Semantic Web can support construction project information management from many perspectives, as outlined in Section 2.3.2. This section presents three possible scenarios to demonstrate how supportive SWIMS could be.

5.3.1 Scenario 1 - Document Management Application

The development of computer networks have resulted in the fact that, in most cases, electronic documents have become the primary media for manipulating and sharing project information between project team members. In a construction project, partners have to manipulate numerous electronic documents (hereinafter referred to as "documents"), such as drawings, reports, and specifications. Although much progress has been made in managing documents in a single organisation, there is still considerable manual work involved in sharing documents between diverse organisations, mainly because of independently developed information management systems and the diverse applications adopted by different parties. Figure 5-3 shows how project documents can be more appropriately managed within SWIMS.

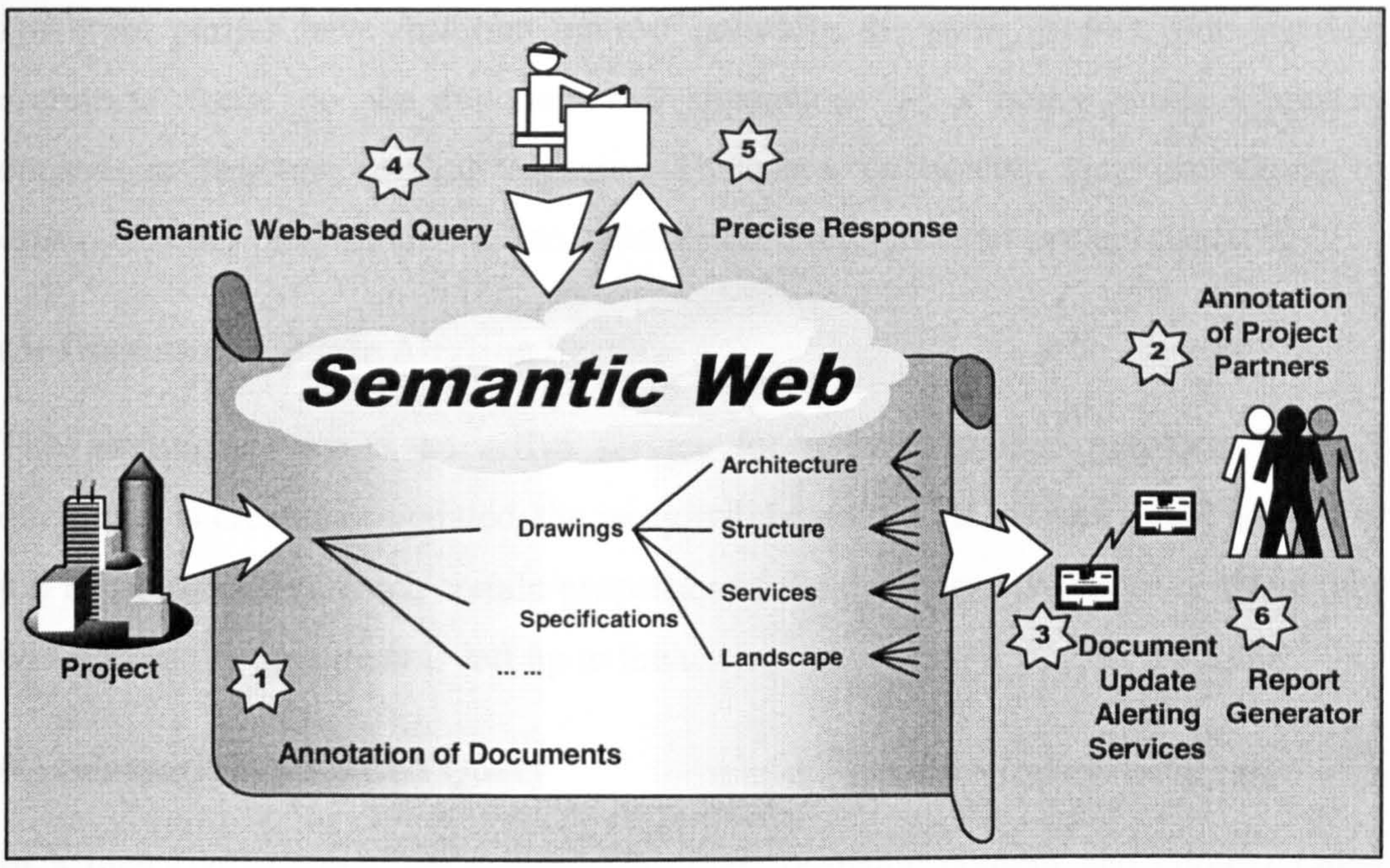


Figure 5-3: Scenario 1 - Document Management

The numbers in Figure 5-3 are explained in details as follows:

(1) Annotation of Documents

Project documents are annotated by metafiles in SWIMS. The metafile is a file which records individual document properties/attributes: the document title, Uniform Resource Identifier (URI), category, creator, "is-about concepts", brief, and version to name a few. The relationships between documents are represented by the ontology of document properties, such as the hierarchy of "is-about concepts". Please refer to Section 5.4 and Chapter 7 for details of document's metafile and ontology. Documents are identified by their properties (within metafiles) and ontology. The metafiles and ontology are processable by both computers (in the format of XML language, RDF statements and OWL language) and human beings (through interface applications). Therefore, both computers and human users can identify the document content by browsing its metafile without reading the document itself.

(2) Annotation of Project Partners

Each project partner has a user profile (a type of metafile), which records a person's properties, such as name, ID, job title, responsibility, priority and interests.

Different parties have different interest points in the same project. For instance, architects focus on the function and appearance of a house while engineers emphasize structure loading capacity. Therefore, documents they use would be different. User profiles are used to annotate the preferences of project partners.

(3) Document Update Alerting Service

The alerting service is an active service for relevant project partners. When a document is created or updated, the party will be notified if his individual interest or job responsibilities match certain properties of the document. Whether or when they will access the document is left up to the users.

(4) Semantic Web-based Query

In a query, documents are accessed by their semantic content and context rather than just by keywords. Enquirers can specify their query by selecting searched domain, terms, corresponding properties and the value of properties. Therefore, the search engine would get the exact meaning of searched terms. For example, computers can understand the real meaning of a term such as *concrete* by identifying that it is discussed in a *material context* or a *construction technology context.*

(5) Precise Response

The computer lists relevant documents in response to a specific query. Documents are retrieved based on query requirements and the enquirer's preference, which are specified within the Semantic Web environment. Clicking on a document title in the result list, related information will be shown and clicking on the URI (a property) of a document will result in the downloading of the file, which may require the user to get authorisation from the corresponding information source. Since the query service is on a semantic-aware basis, the precise response can improve efficiency of accessing documents from heterogeneous repositories for project partners.

(6) Report Generator

When the structure of a report is well-defined, SWIMS can extract related information from the Data Repository and generate the report automatically. This is useful when the project information updates frequently.

Therefore, project documents can be more appropriately managed within SWIMS and the problems of accessing and updating information in diverse databases can be resolved. The advantages include:

(i) Avoiding the problem of accessing documents written in various formats, terminologies, language and standards.

(ii) Saving time involved in responding to a specific query.

(iii) Reducing the manual work in identifying the most relevant documents from partner's information repositories.

(iv) Enhancing the accessibility of content and protecting safety of documents.

(v) Updating report automatically from dynamic information sources.

5.3.2 Scenario 2 - Product Specification Conversion Application

The first scenario demonstrates how the Semantic Web can support the document sharing, retrieval and generation. The following scenarios demonstrate how document sharing could facilitate collaboration between project partners.

Two cases are presented in this scenario to show the conversion between different product specifications using SWIMS. Figures 5-4, 5-5 and 5-7 depict a project of constructing a bridge in Country A, which involved project team members from different companies in various countries. Here, A stands for America, B stands for British (U.K.), C stands for China, and E stands for Europe Union. The collaboration between globally distributed team members requires an efficient information sharing approach.

Case 1: Measurement System Conversion

In the construction of the bridge, cables composed of galvanized bridge wires were needed. Supposing there were 110 steel cable suppliers in the supplier catalogue, 10 of which can provide the required materials. The products were made according to suppliers local specifications (illustrated in Figure 5-4 and Table 5-1).

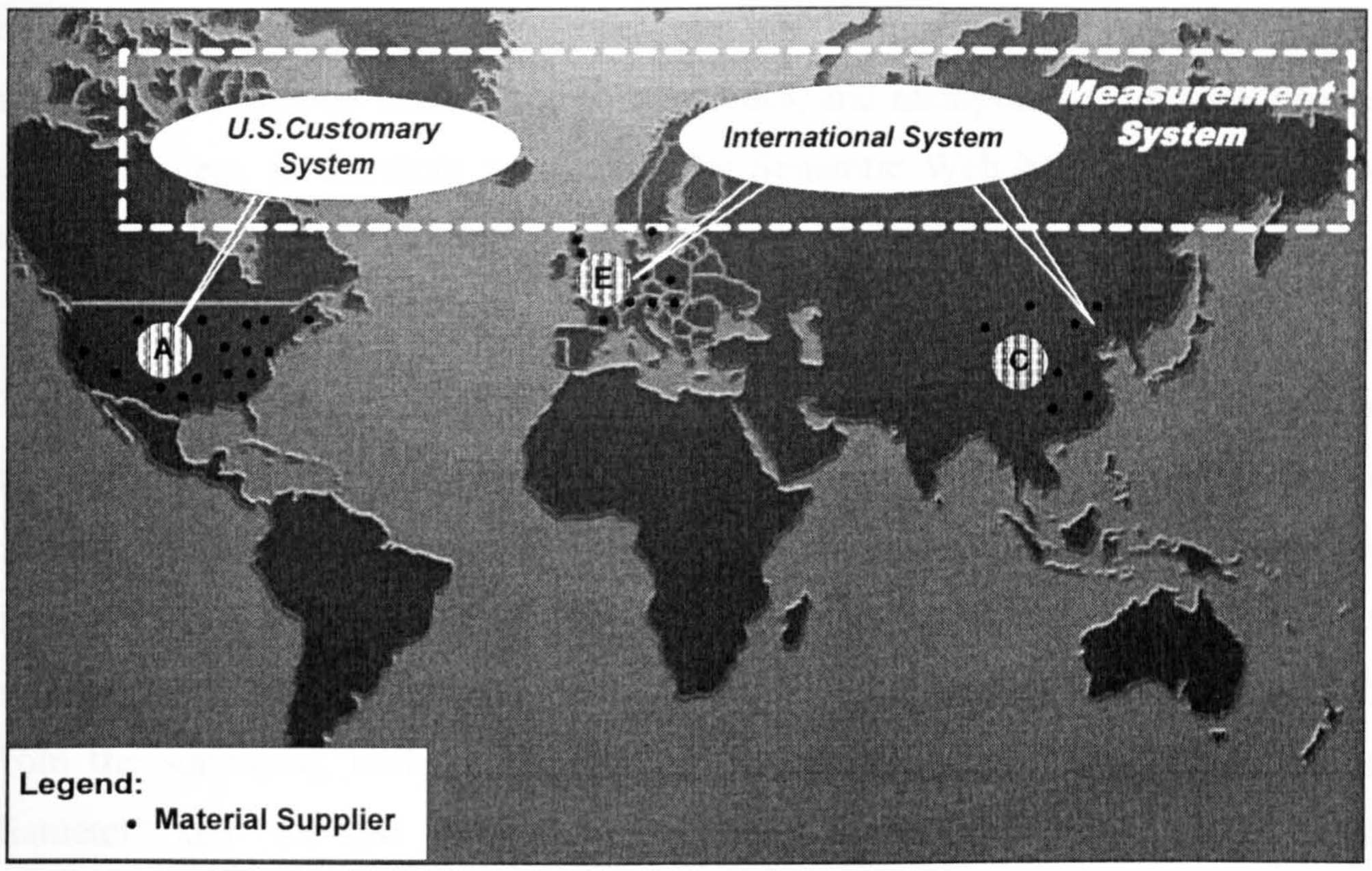


Figure 5-4: Scenario 2a - Measurement System Conversion

Table 5-1: Cable Suppliers List

Countries	No. of Suppliers	No. of Specialist Suppliers	Measurement Systems
A	79	4	U.S. Customary System
E	24	4	International System
C	7	2	International System
Total	110	10	

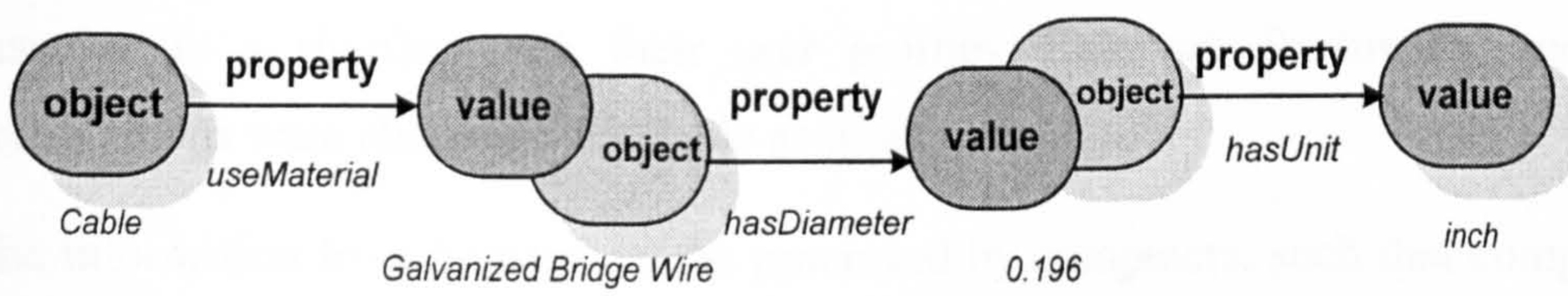
The design documents, material information and supplier profiles were stored in Data Repositories, e.g. structural engineer drawings, product databases and supplier Websites. The project manager was a member of Personnel who issued the query/RFI using a search engine.

In the framework of traditional electronic information management systems, as depicted in Figure 5-1, if the keywords of "cables composed of 27,572 #6 galvanized bridge wires with 0.196 inch in diameter" were inputted into the current Web-based search engine, the computers would not know that a cable with "2.54 centimetres in diameter" is the same as one of "1 inch in diameter", and that the wires can be grouped in either "61 strands of 452 wires each" or "50 strands of 552 wires each". Therefore, the search result was incomplete. To get a precise answer, a manual search was required. Obviously, the less precise the answer is, the lower the

efficiency of the construction process is. An innovative approach, which can aggregate information from different companies, and manipulate it by both humans and computers, is therefore necessary. The Semantic Web has been proposed as such a solution.

Conducting the above search in the framework of SWIMS, which is depicted in Figure 5-2, a query was made by the project manager (the Personnel), who inputted his requirements into the search engine (the Interface Application) as "the type of cable is made of a large number of individual galvanized bridge wires which are parallel to one another. The recommended diameter of wires is 0.196 inch". An information extraction software collected the RFI and extracted the key concepts from the statement, such as "material is galvanized bridge wires 0.196 inch in diameter" and "the unit is in U.S. Customary System". The XML/RDF Parser represented the concepts using RDF as shown in Listing 5-1:

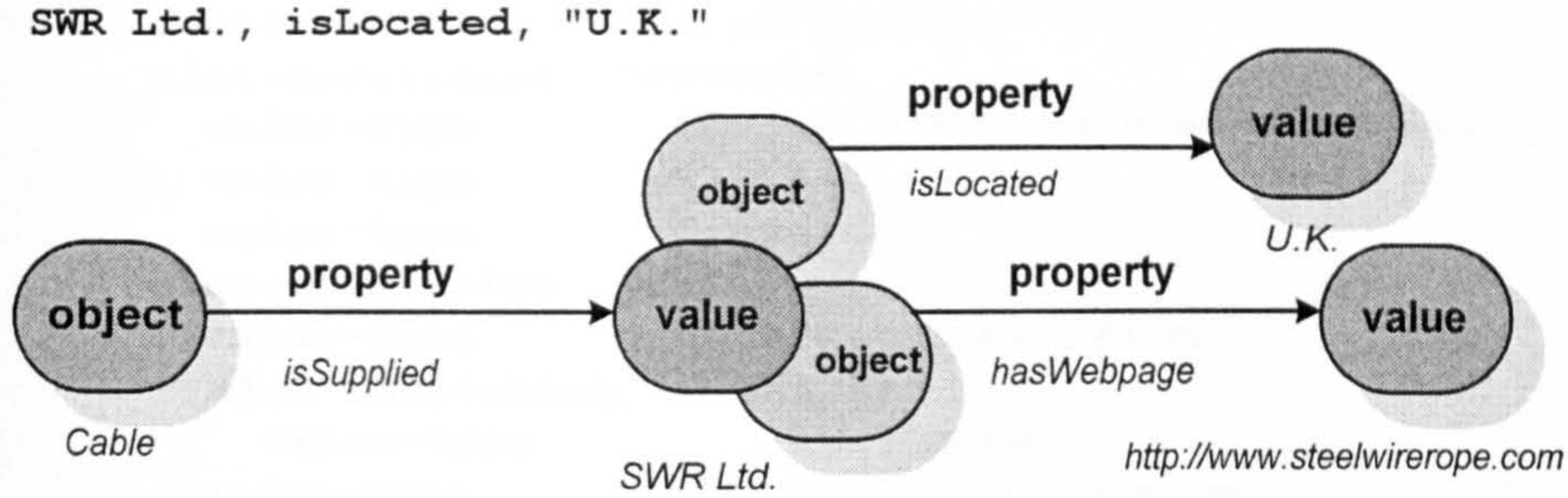
```
Cable, useMaterial, "Galvanized Bridge Wire",
Galvanized Bridge Wire, hasDiameter, "0.196"
0.196,hasUnit,"inch"
```



Listing 5-1: RDF Annotation for Cable Case

These metadata of RFI were sent to agents, the Material Domain Agent and Measurement Conversion Co-operation Agent, for possible matching. Responding to the RFI, the supplier catalogue in the Data Repository was retrieved and the product information were parsed and sent to the Material Agent in RDF triples as well, as shown in Listing 5-2.

```
Cable, isSupplied, "SWR Ltd.",
SWR Ltd., hasWebpage, "http://www.steelwirerope.com"
SWR Ltd., isLocated, "U.K."
```



Listing 5-2: RDF Annotation for Cable Supplier

The Agents then compared the metadata of RFI with products. The Co-operation Agent converted the data in the International Measurement System to the U.S. Customary System. Using the RFI and product information in the same measurement system, the Material Agent inferred the rules and found matches. For example, as shown in Listing 5-3, wires that were 0.47-0.51 centimetres in diameter (1 inch=2.54 centimetres) were compatible with the requirement (error within 5%). All matched suppliers (10 companies in this case) then were reported to the project manager in a shortlist with their user profiles. Relevant documents, such as certifications were also made available as proofs.

The information from humans can be processed by computers, such that computers can convert data between the U.S. Customary System and the International System and find compatible product by logical reasoning. As a result, appropriate companies were retrieved automatically and accurately, overcoming the potential overload or loss of information using conventional approaches. This enhances the efficiency of the construction process and meets user demands.

```
class-def  specifications %specifications are a class
class-def  measurement
  subclass-of        specifications
  slot-constraint    hasMember
    value-type         Length  %that is a measurement
    value-type         Width
    value-type         Diameter
  slot-constraint    useSystem
    value-type         International System
     slot-constraint     hasUnit
       value-type          centimeter
    value-type         U.S. Customary System
     slot-constraint     hasUnit
       value-type          inch
       slot-constraint     isEqualTo
         value-type          2.54
         slot-constraint     hasUnit
           value-type          centimeter
class-def   bridge cable
  subclass-of        NOT specification
  slot-constraint    isMadeOf
    value-type         galvanized bridge wires
    slot-constraint    hasDiameter
      value-type         0.196
        slot-constraint    hasUnit
          value-type         inch
      slot-constraint    isEqualTo
        value-type         0.49
        slot-constraint    hasUnit
          value-type         centimeter
      slot-constraint    error
        value-type         5%
class-def   company
  subclass-of        NOT specification
  slot-constraint      hasMember
    value-type           SWR Ltd.
    slot-constraint      isLocated
      value-type           U.K.
      slot-constraint      useSpecification
        value-type           International System
    slot-constraint      produce
      value-type           No 6 wire
      slot-constraint      hasDiameter
        value-type           0.50
        slot-constraint      hasUnit
          value-type           centimeter
```

Listing 5-3: Partial Ontology for Cable Case

Case 2: *Technical Standard Conversion*

In this project, the project manager (in Country A) wanted to use some steel structural elements from Country C on the project in Country A because the steel price was much lower. Therefore, the designers (in Country A) needed to use product information from Country C. However, the technical standards relating to

structural steelwork in the two countries were different (as shown in Figure 5-5). Many designers have difficulties converting between standards in different countries and this problem is compounded when they are written in different languages. Professionals were required to deal with this and much manual work and time will be consumed in searching for the appropriate information. Fortunately, technical standards and languages are convertible in the SWIMS. Extracting technical standards and product information from the Data Repository, using a Co-operation Agent with appropriate ontologies, computers can identify that the steel with grade Q345 in GB T700-1988 standard (in Country C) had similar structural properties as the one with grade Gr.50 in the ASTM standard (in Country A). Therefore, searching for heterogeneous information is made much easier. Figure 5-6 shows a slightly simplified version of the ontology for converting steel standards between Countries A and C.

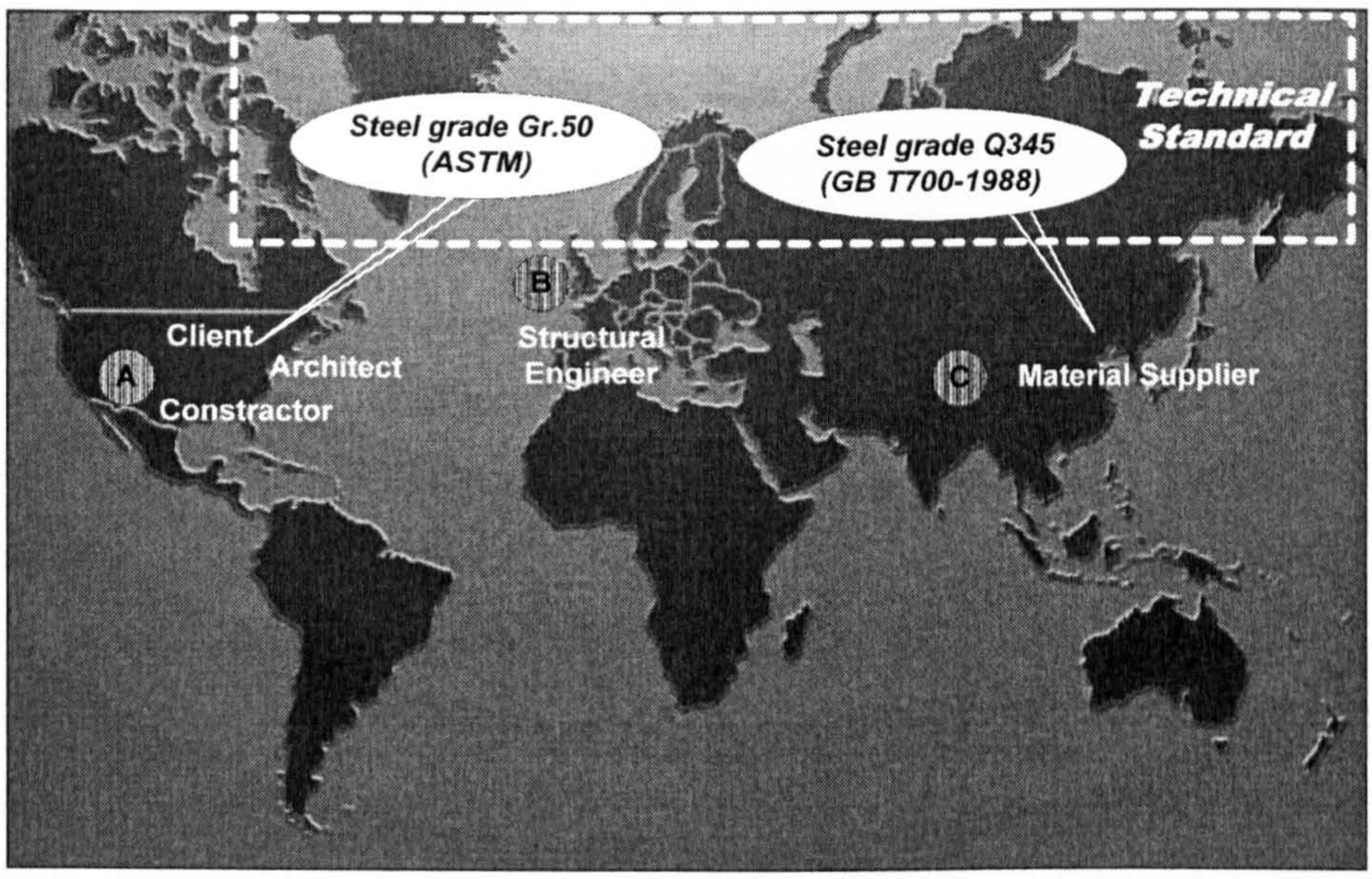


Figure 5-5: Scenario 2b - Technical Standard Conversion

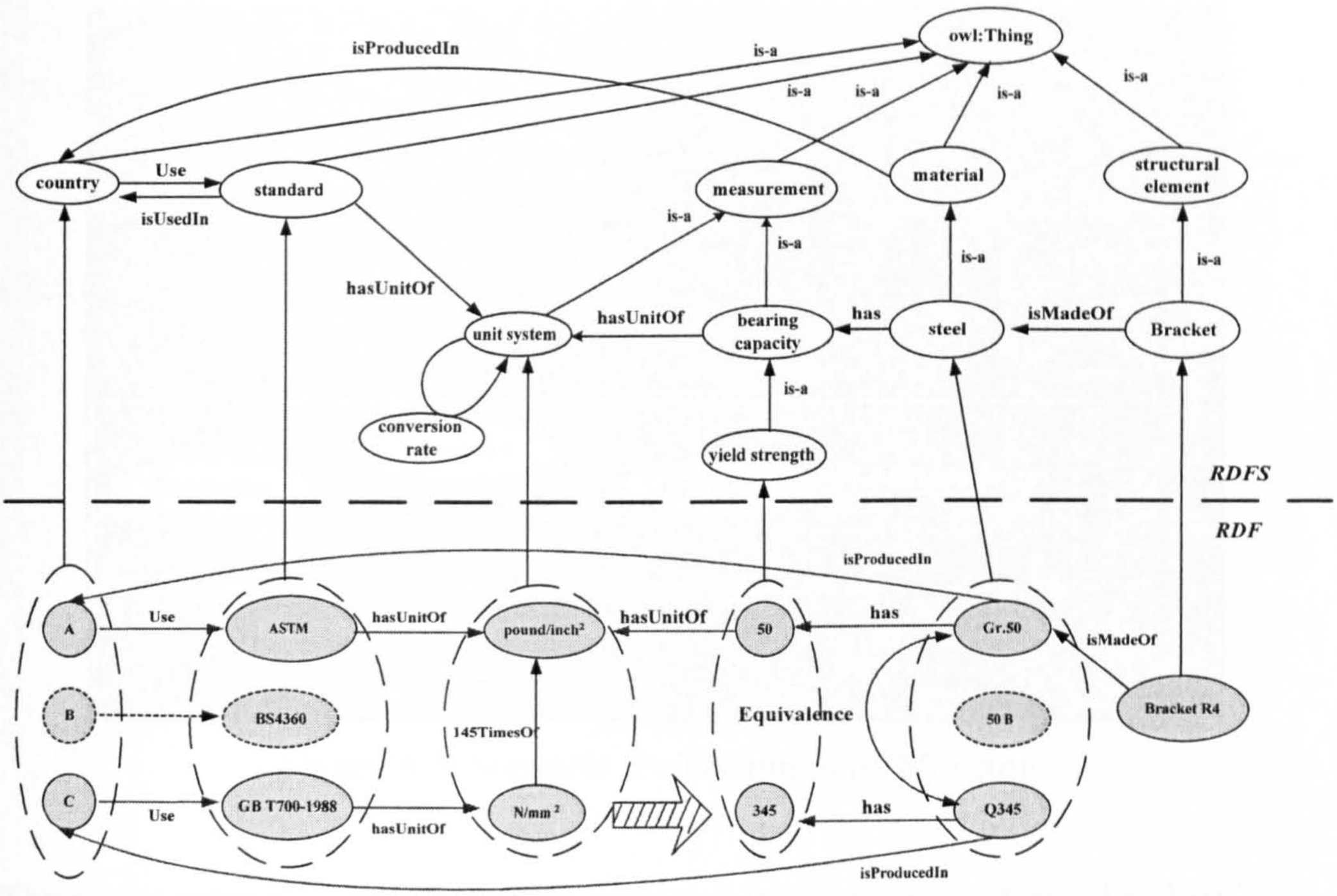


Figure 5-6: Standard Conversion Ontology Represented in RDF(S) and RDF

5.3.3 Scenario 3 - Terminology Merging Application

In the previous project, an object may have various names in different professional fields. For instance, a structural element may be called "the support of console attached to a wall" by a client, "a bracket" by an architect, or "a half beam" by a structural engineer (as illustrated in Figure 5-7). SWIMS could translate these terminologies. These terms were defined as synonymous in the design ontology. Objects described by the client, engineer and architect can be identified as the same object if the value of each object's properties were matched. Please refer to Chapter 7 for the demonstrator of this case. Therefore, even if the team members cannot communicate in person, they can still obtain the right information from a partner's data repository with the help of SWIMS.

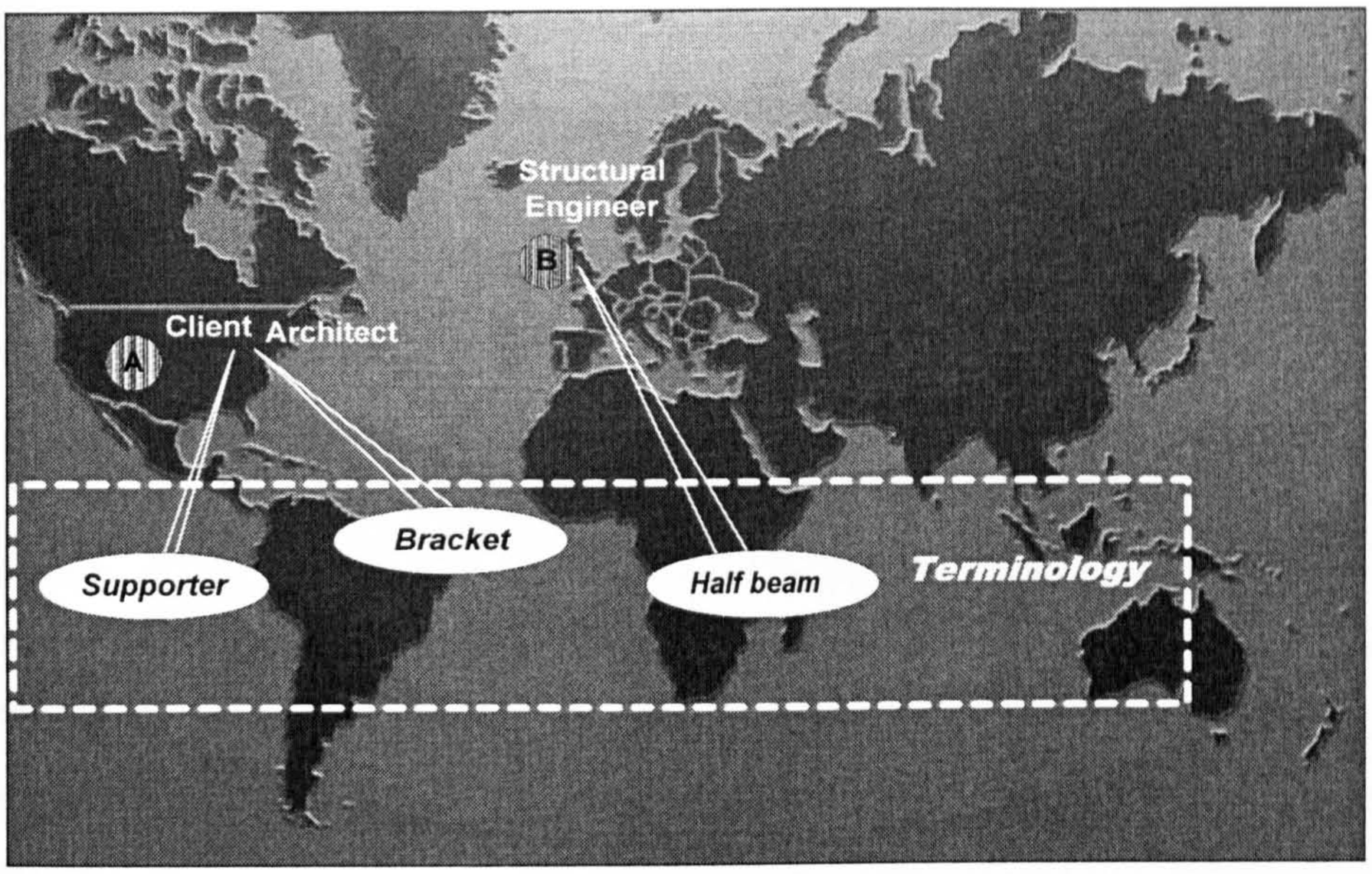


Figure 5-7: Scenario 3 - Terminology Merging

The above scenarios demonstrate the potential deployment of SWIMS. The key issue in achieving them is to represent both the content and context of construction information in a computer processable way. In Sections 5.4, 6.5 and 6.4, annotation models and ontologies are presented to explain how construction information can be interpreted semantically. Several demonstrators are displayed in Chapter 7.

5.4 ANNOTATIONS AND METAFILES

5.4.1 Annotation of Documents in Other Industry Sectors

Annotation means "comments, notes, explanations, or other types of external remarks that can be attached to any document or a selected part of the document without actually needing to manipulate the document. When the user gets the document he/she can also load the annotations attached to it from a selected annotation server or several servers and see what his/her peer group thinks" (Annotea Project, 2005). In the Semantic Web environment, computers will identify the information in a document by parsing its annotation. The experiences of document annotation from other industry sectors provide rich references for the annotation of construction project information.

An example in annotation is the **Rich News** system (Dowman et al, 2005). It automatically annotated radio and television news from the World Wide Web. Using RDF annotations, it identified entities referred to in text documents. These

Entities are organised using an ontology with three modular layers, 250 classes and 100 properties. Most news stories were annotated as shown in Figures 5-8 and 5-9, containing: a Headline, a Section slot (classification of the story e.g. sports, politics, British and London), a Description slot (brief summary), a Words slot (keywords), and StartTime and EndTime.

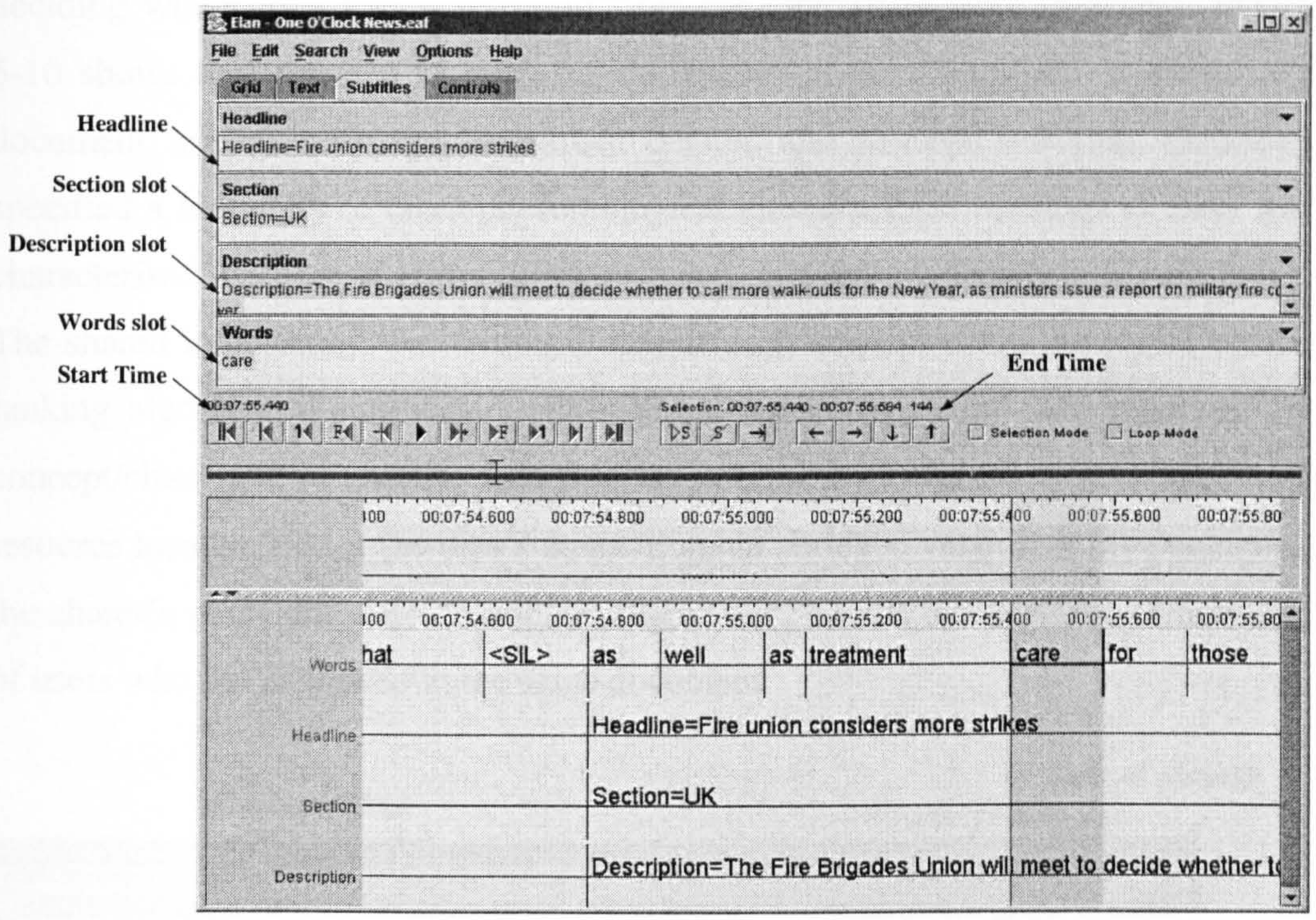


Figure 5-8: Annotations on a Radio Broadcast (Dowman et al, 2005)

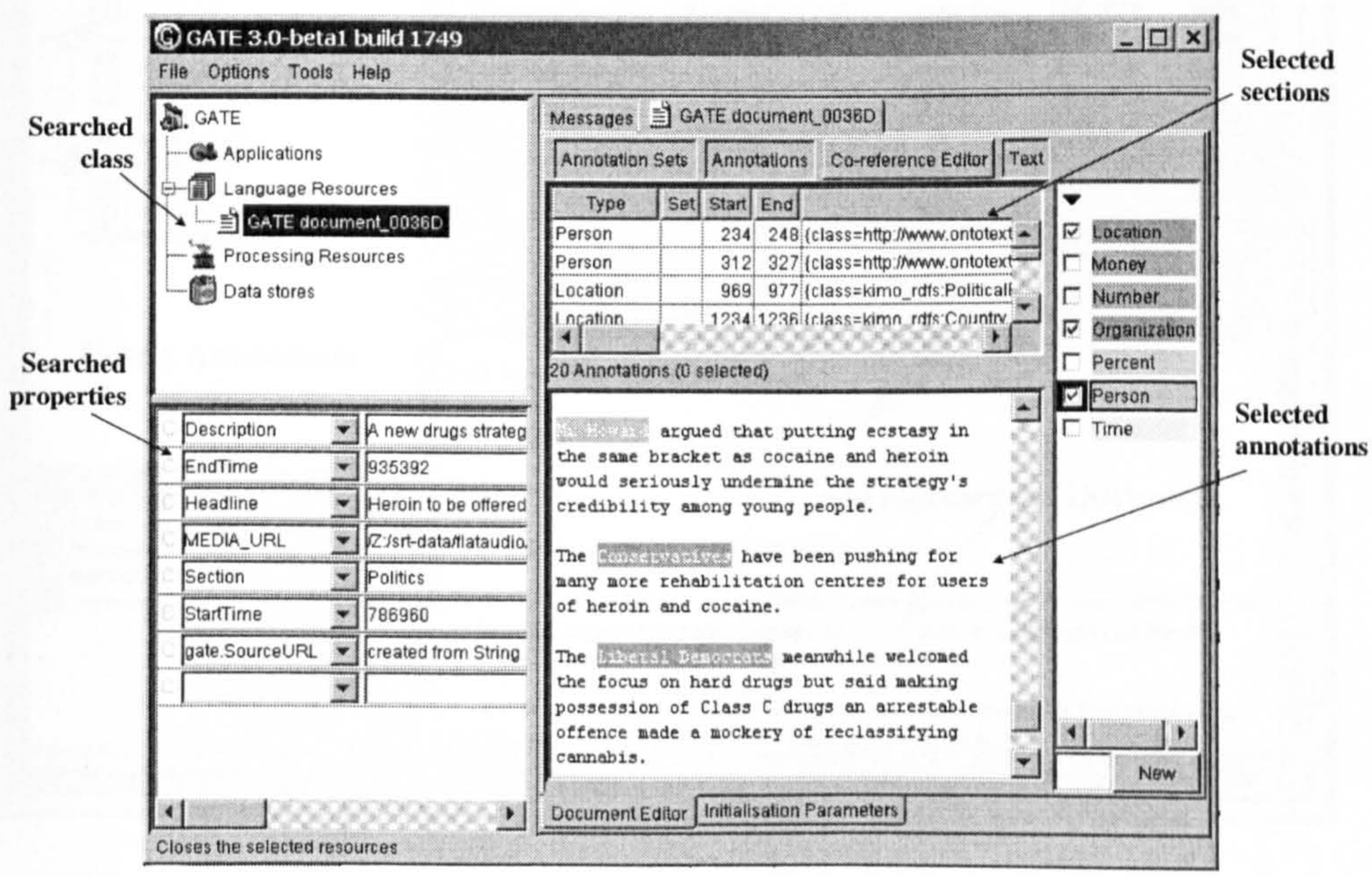


Figure 5-9: Index of an Annotated Document (Dowman et al, 2005)

Another example is **OntoShare**, which is an ontology-based knowledge sharing software (Davies et al, 2003). Documents in OntoShare were annotated by a set of ontological concepts and comments on the information of the document. Annotations were stored with documents. This was very useful for other users in deciding which information retrieved from the OntoShare store to access. Figure 5-10 shows the interface of indexing documents in the OntoShare. It included a document list, an ontology, document details, and concept selection. Ontology specified a hierarchy of concepts (ontological classes). Each ontological class was characterised by a set of terms (keywords and phrases) (as shown in Figure 5-11). The shared information was matched against each concept using the vector cosine ranking algorithm (Granovetter, 1982). Each document shared was annotated by concept/class and properties, including keywords, a document title, a universal resource locator (URL), the user's annotation, an abridged version of the document, the sharer's name and date of storage. Figure 5-12 shows the interface of indexing of users who are interested in the same document.

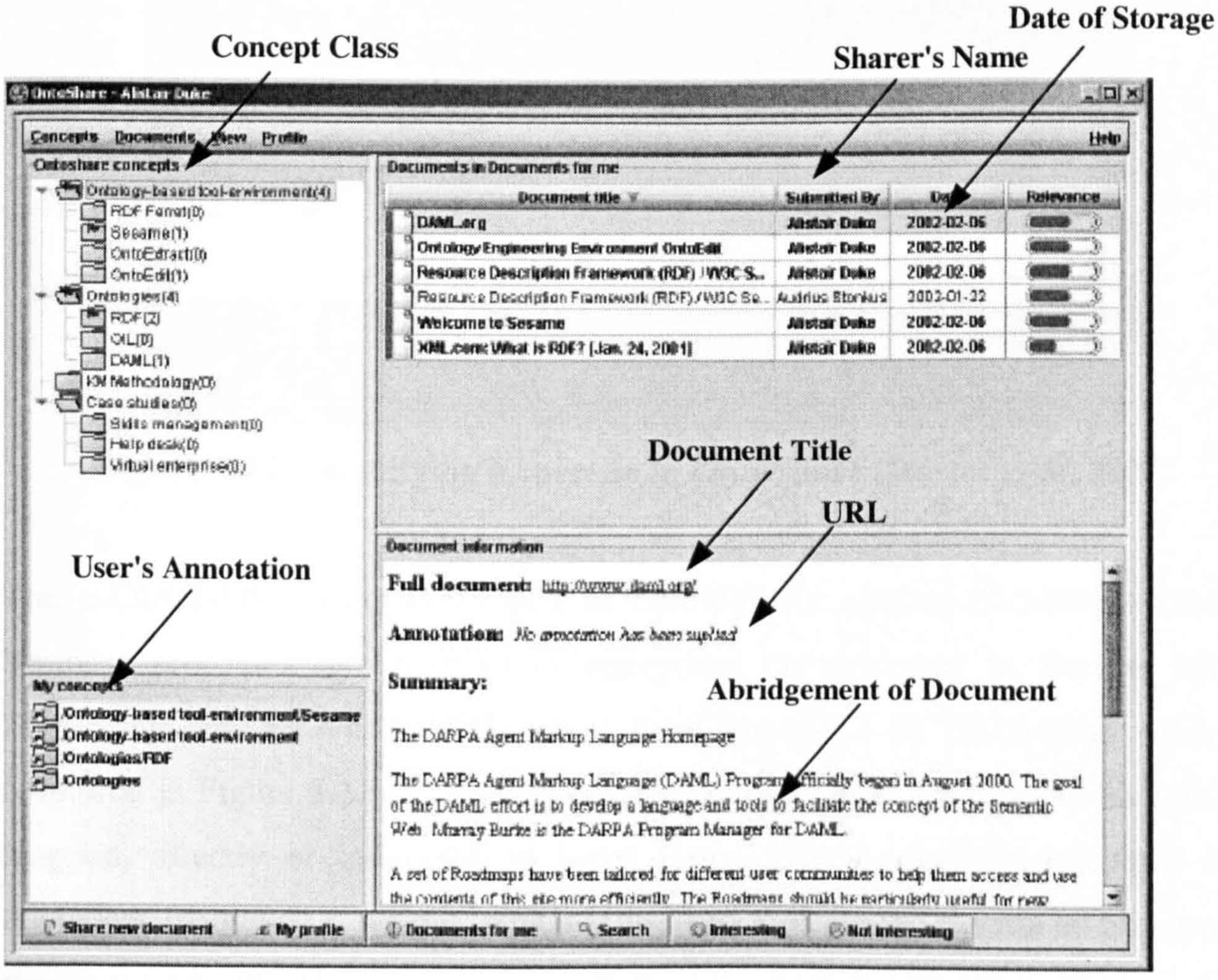


Figure 5-10: Listing Annotated Documents in OntoShare (Davies et al, 2003)

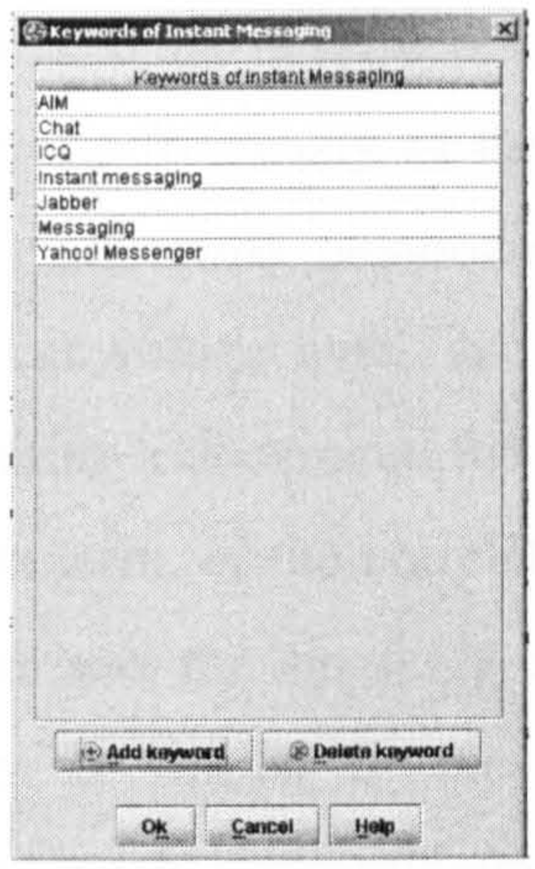


Figure 5-11: Annotating Ontological Keywords to a Document in OntoShare (Davies et al, 2003)

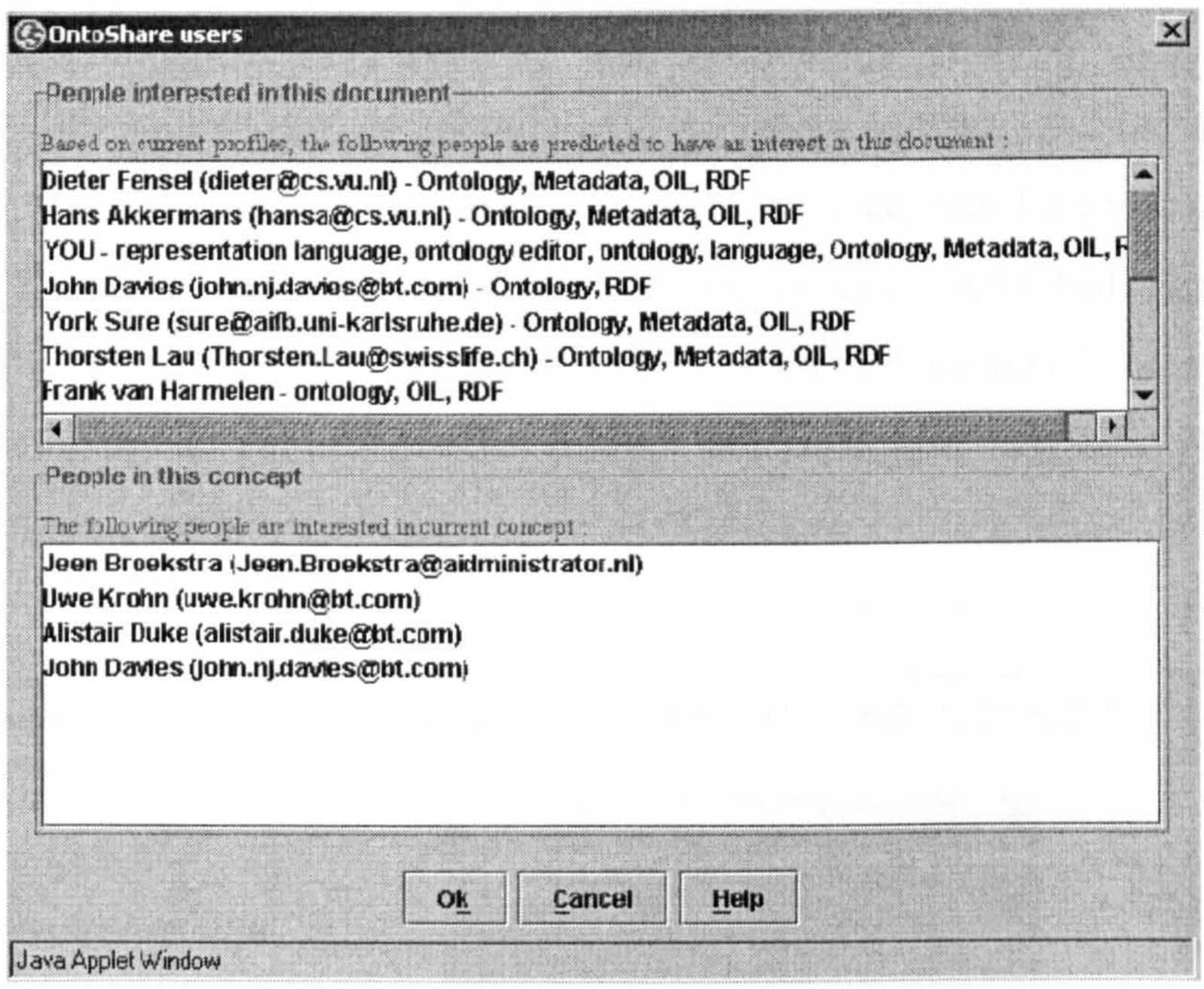


Figure 5-12: Identifying Expertise in OntoShare (Davies et al, 2003)

The **e-COGNOS** project developed an ontology for sharing documents used in business processes by construction enterprises (as reviewed in Section 2.3.2). Documents on the e-COGNOS server were annotated by ontological concepts (captured in Figure 5-13). Each concept is associated with a weight number and is mapped, directly or indirectly, to index terms. Every concept/index term of a document is granted a weight, which is used to indicate the degree of importance the concept/index term has over a document. A thesaurus is used to specify synonyms for each term and map indirectly index terms to a concept. The weight of

a term to a document is calculated by several parameters according to a vector model. The parameters include the frequency of the index term mentioned in the document, the maximum over all terms mentioned in the document, the total number of documents in the knowledge base, and the number of documents in which the term appears. Annotating a document, the weight of a concept is decided by the weight of its direct index term, or the correlation factor (cosine of the angle between the index term vector and the ontology concept vector) applied to its indirect index. This algorithm is similar to the one of Ontoshare, in which information is matched against each concept/keywords using the vector cosine ranking algorithm. Figure 5-13 shows the interface for submitting and annotating a document on the e-COGNOS server. It includes three parts: document submission window, document details window, and document's concept window. The document submission window consists of a document location and the document language. The document details window consists of document location, document language, document tile, subject, author, version, revision, status and number. The document's concept window consists of a concept list slot, an added concept slot, a slot indicating the weight of the added concept and a slot indicating the ontology the added concept is from.

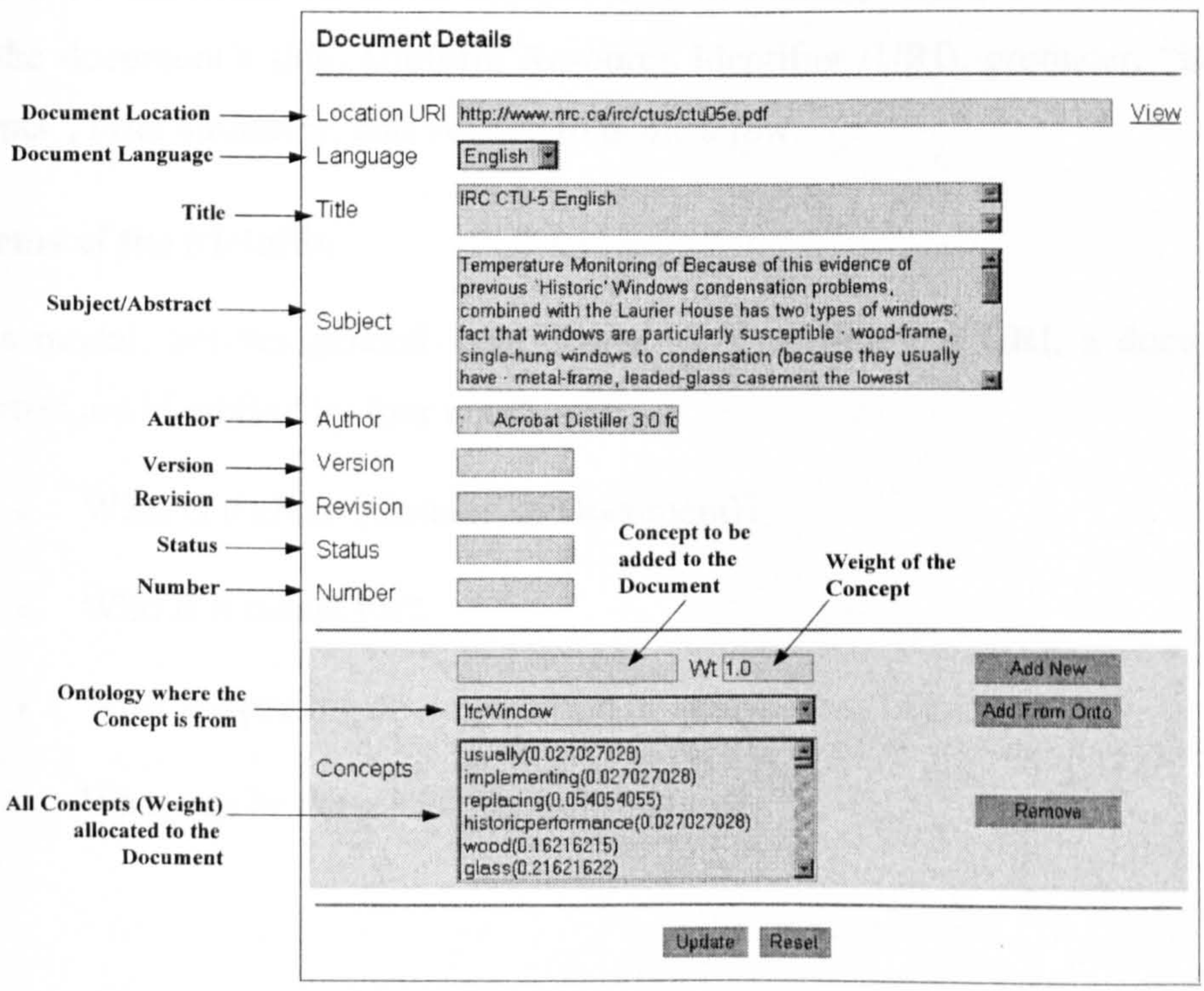


Figure 5-13: Document Annotation on e-COGNOS Server (e-COGNOS Consortium, 2002)

The above examples imply that a document's annotation consists of two types of slots: traditional slots and semantic slot. Taking the annotation on e-COGNOS Server as an example, the traditional slots include the document's title, language, location, author, version, etc. The semantic slots include the concept slot, ontology slot and weight slot. It is the semantic slots that enable computers to access the conceptualised information contained in a document.

5.4.2 Annotation of Construction Project Documents

Project documents (specifications, drawings, Webpages, etc.) are annotated using metafiles in SWIMS. A metafile is a file which records an individual document's attributes. A document's attributes include concepts and properties matched to the information in the documents. An information management task is performed based on these annotations and the ontology associated with the task. Ontologies can penetrate the boundaries of various applications and data repositories, so as to make information more accessible. An annotation model for documents is proposed here.

(1) Metafile Model

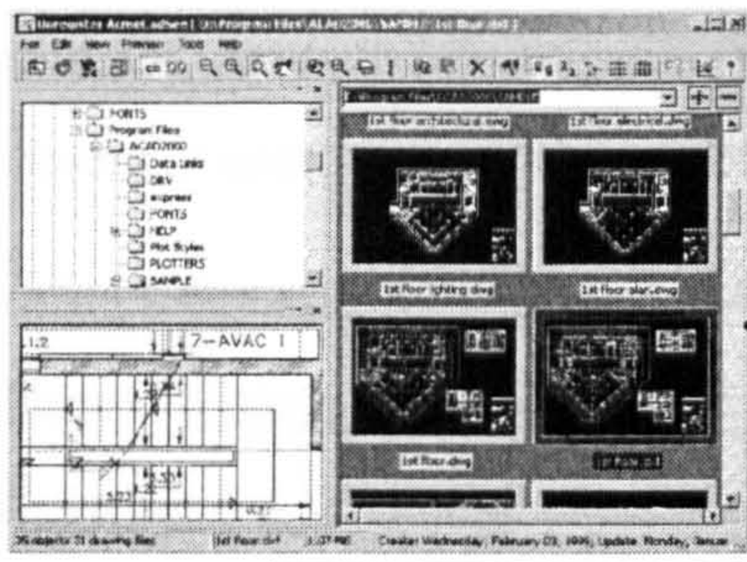
Figure 5-14 illustrates the basic metafile of a document, which is a tabular model with the document's title, Uniform Resource Identifier (URI), producer, "is-about concepts", brief summary, and version to name a few.

(2) Items of the Metafile

In this model, besides general information, such as title and URI, a document's properties are identified by four questions:

- What is it about (Content of Document)?
- Who is it meant for?
- What supporting documentation or analysis has been used?
- What can be done with the information?

Document



Metafile

Properties	Values
Title	#Literal (XML)
URI	#Literal (XML)
In Category Of	#Document Type(Class)
Is About	#Material or Structural Element or Activity Zone (Class)
Is Characterised By	#Literal (XML)
In Project Of	#Project List (Class)
In Format Of	#Present Medium (Class)
Use Software	#Literal (XML)
Brief Summary	#Literal (XML)
Comment	#Literal (XML)
Authority	#Organization or Actor (Class)
Is Produced By	#Organization or Actor (Class)
Date Produced	#Date (XML)
Is Used By	#Organization or Actor (Class)
Last Accessed By	#Organization or Actor (Class)
Date Last Accessed	#Date (XML)
Version	#Version (Class)
Is Refer To	#Document (Class)
Include Document	#Document (Class)
Depend On	#Document (Class)
Is Belong To	#Document (Class)
Is Applied In	#Activity Zone or Process Stage (Class)
In Conjunction With	#Document (Class)
Is Used To Generate	#Document (Class)

Figure 5-14: Document Metafile

(i) "**General information**" consists of permanent properties, which are specified with the creation of the document and will not change in most situations. "General information" contains:

Title: the literal expression of the document's name.

URI: the Uniform Resource Identifier of the document. This is used to identify and access the document.

(ii) "**What is it about**?" asks for the project information contained in the document. "What is it about?" contains:

In Category Of: specifies the subject content of the document, which is defined by the enumeration of *project administration*, *technical*, *legal* and *commercial*. The types of document content have been reviewed in Section 3.6.1-(3).

Is About: outlines the ontological concepts that the document relates to. The range covers the classes: Material, Structural Element and Activity Zone. Concepts in the Material and Structural Element classes adopt IFC definitions, which have been reviewed in Section 3.6.3-(3). The Activity Zone adopts the definitions from the Process Protocol project, which has been reviewed in Section 3.6.1-(1). Please refer to Section 6.5-(2)-Step four for more details about each class.

Is Characterised By: lists the keywords associated with the document. There are three ways to specify keywords:

- Firstly, the document has been annotated by several concepts and each concept class is characterised by a set of keywords. These keywords are then allocated to the document.

- Secondly, applying a vector model to extract keywords from a textual document. The vector model was originally adopted by the e-COGNOS project as an algorithm to calculate the weight of a concept to a document. The SWIMS document annotation model presents more semantic features of a document than the e-COGNOS document annotation model. One of them is a keyword. Keywords, as in traditional document management systems, can be used in a document index and search. Additionally, in the SWIMS document management system, it plays an intermediate role between documents and concepts. $KeywordIndex_{i,j}$, the index of keyword i to document j, indicates the relative frequency of keyword i being used in document j. It is calculated by Equation (5-1). Thus, users can select keywords from the rank of $KeywordIndex$.

$$KeywordIndex_{i,j} = NormFreq_{i,j} \times InverseFreq_i \qquad \textbf{(5-1)}$$

  Where $NormFreq_{i,j}$ represents the normalised frequency of the keyword i in the document j, which is calculated using Equation

(5-2), and $InverseFreq_i$ represents the inverse of the frequency of keyword i among all documents in its domain information repository, which is calculated using Equation (5-3).

$$NormFreq_{i,j} = \frac{Freq_{i,j}}{\max_l Freq_{l,j}} \tag{5-2}$$

Where $Freq_{i,j}$ represents the frequency of keyword i mentioned in document j, and $\max_l Freq_{l,j}$ represents the frequency of keyword l, which is the maximum frequency of all keywords mentioned in document j.

$$InverseFreq_i = \log \frac{N}{n_i} \tag{5-3}$$

Where N represents the total number of documents in its domain information repository, and n_i represents the number of documents in which keyword i appears.

Equations (5-1) to (5-3) are quoted from the e-COGNOS project (Mezian and Rezgui, 2004).

- Thirdly, users specify keywords directly. These keywords are compared to each concept's keywords. If they match strongly enough, the document would suggest linking to that concept. If the suggestion is accepted, the remaining keywords of the concept will be allocated to the document refining its annotation.

Whichever approach is applied, users make the final decision on the allocation of keywords to a document.

In this research, the term Mate is used to define such a relationship that a keyword and a concept are used to annotate the same document. The index $Mate_{i,j}$ represents the frequency of keyword i and concept j being used together in annotation. Normally, a concept and its associate keywords are mates. If concept j has k keywords in the ontology, and has another mate keyword i, which is different

from k. Then, $IndexMate_{i,j}$ in Equation (5-4) is used to identify if the keyword i should be linked to concept k as one of its keywords.

$$\begin{cases} IndexMate_{i,j} = \dfrac{Mate_{i,j}}{StaMate_j}, \text{ if } StaMate_{i,j} \neq 0, \\ IndexMate_{i,j} = 1, \text{ if } StaMate_{i,j} = 0 \end{cases} \quad \textbf{(5-4)}$$

$$\text{Where, } StaMate_j = \frac{1}{k}\sum_k Mate_{k,j} \quad \textbf{(5-5)}$$

Where, $StaMate_{i,j}$ is the mean of all $Mate_{k,j}$, as calculated in Equation (5-5). If $IndexMate_{i,j} \geq 1$, then keyword i will be added to the ontology as a keyword of concept j.

The similarity of keywords can be used to identify related concepts. Supposing concept A has i keywords, concept B has j keywords ($i \leq j$), if more than $\text{int}\frac{j}{2}$ keywords of A and B are same, A and B will be suggested to be related concepts by computers if they have no defined relationship yet. Their relationships could be *sub-class-of* or *the same as*. Humans will make the final decision.

In Project Of: specifies the project in which the document is used.

In Format Of and **Use Software**: specify the presentation medium of the document's contents and its associated software, which has been reviewed in Section 3.6.1-(4).

Brief Summary: abstracts the document's contents. Several summarizers are available for this work as reviewed in Section 5.2.3.

Comment: the comments on the document are from users in free text format. The system collects feedback on a document from users, and enables the sharing of knowledge between related project partners.

(iii) **"Who is it meant for?"** consists of the relationships between documents and project partners. It could be used for customised information distribution and retrieval services. "Who is it meant for?" contains:

Authority: indicates the maintainer of the document. It could be the same person as the "Is Produced By".

Is Produced By: indicates the creator/owner of the document; this may be one or more participants in the project.

Date Produced: the date the document was produced.

Is Used By: lists users who are highly interested in the document. This will be one or more project partners, e.g. Actor and Organisation.

Last Accessed By: indicates the last user of the document. It is a sub-property of the "Is Used By". It is a functional property, which is defined in Section 6.5-Step five-(2). For a given document, there can be at most one actor that is related to the document via the "Last Accessed By" property. As shown in Figure 5-15, if *J.Pan* and *Jiayi* are instances/individuals of Actor. Document A is "Last Accessed By" *J.Pan*, meanwhile, Document A is "Last Accessed By" *Jiayi*. Then because "Last Accessed By" is a functional property, we can infer that *J.Pan* and *Jiayi* must be the same individual. If *J.Pan* and *Jiayi* were explicitly stated to be two different individuals, then the above statement would lead to an inconsistency error.

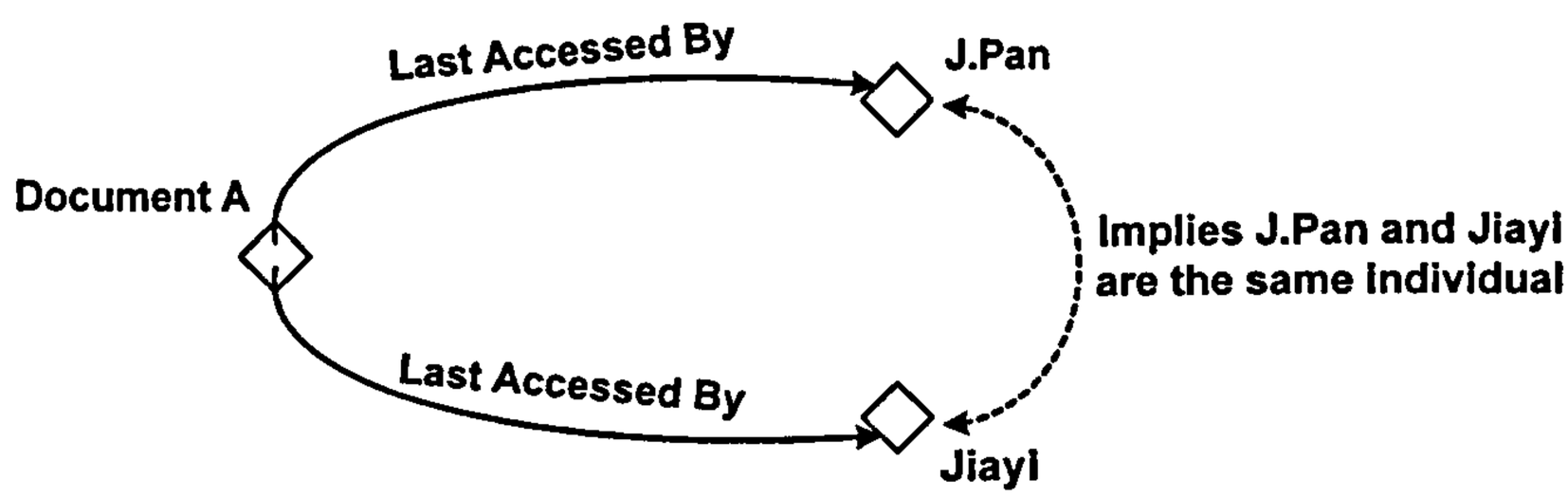


Figure 5-15: An Example of a Functional Property: Last Accessed By

Date Last Accessed: the most recent date the document was used. It is a functional property as well.

(iv) **"What supporting documentation or analysis has been used?"** consists of the project process information and interdependencies between documents. It could be used for information tracking, retrieval, trustworthiness evaluation and context-specific information/knowledge management. "What supporting documentation or analysis has been used?" contains:

Version: the version number and evolution status of the document. The evolution status is defined by the enumeration of *initial*, *revised*, *updated*, and *finalised*.

Refers To: represents the reference relationship to a document. The document refers explicitly to the name or title of the element mentioned in another document. It is a transitive property, which is defined in Section 6.5-Step five-(2). If Document A "Refers To" Element B.1 in Document B and Element B.1 "Refers To" Element C.1 in Document C, then we can infer that Document A "Refers To" Element C.1 in Document C. This is indicated by the dashed line in Figure 5-16. For example, the drawings "Refers to" function definitions in specifications about project techniques and functions.

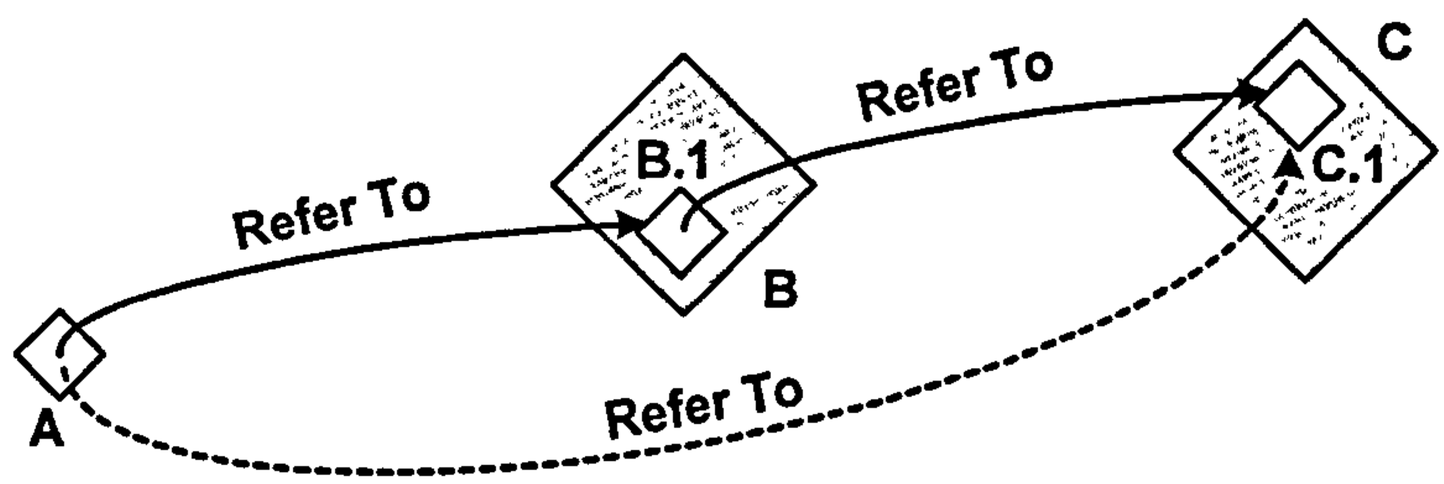


Figure 5-16: An Example of a Transitive Property: Refer To

Includes Document and **Belongs To**: specify the interdependency between a parent document and its child documents or chapters, if the chapter is saved as a separate document from other chapters. The property "Belongs To" is an inverse property of the property "Includes Document", and vice versa. The definition of inverse property refers to Section 6.5-Step five-(2). For example, if Document A "Includes Document" Document B, then we can infer that Document B "Belongs To" Document A. This is indicated by the dashed line in Figure 5-17. Both "Belongs To" and "Includes Document" are transitive properties.

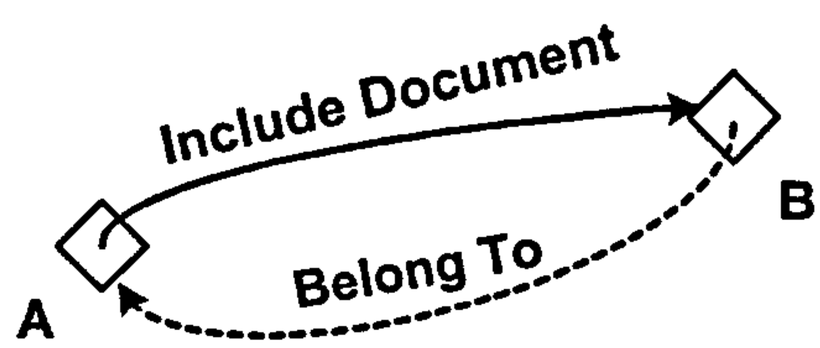


Figure 5-17: An Example of an Inverse Property: "Include Document" has an inverse property that is "Belong To"

Depends On: represents the relationship between a document and its implicit references. The document is semantically related to the element mentioned in a reference though an explicit quote is not made. It is a complement of the "Refer To".

(v) **"What can be done with the information?"** consists of the project process information and interdependencies between documents as well. It could be used for information distribution, rule inference, and content-based information/knowledge management. "What can be done with the information?" contains:

Is Applied In: specifies the Activity Zone and Process Stage in which the document is applied. The Activity Zone and Process Stage adopt the definitions in the Process Protocol project, which has been reviewed in Section 3.6.1-(1).

In Conjunction With: specifies the documents that are often combined with this one to take effect in a task. It is a symmetric property, which is defined in Section 6.5-Step five-(2). Figure 5-18 illustrates an example of the symmetric property. If Document A is related to the Document B via the "In Conjunction With" property, then we can infer that Document B must also be related to Document A via the "In Conjunction With" property. In other words, the property "In Conjunction With" is its own inverse property.

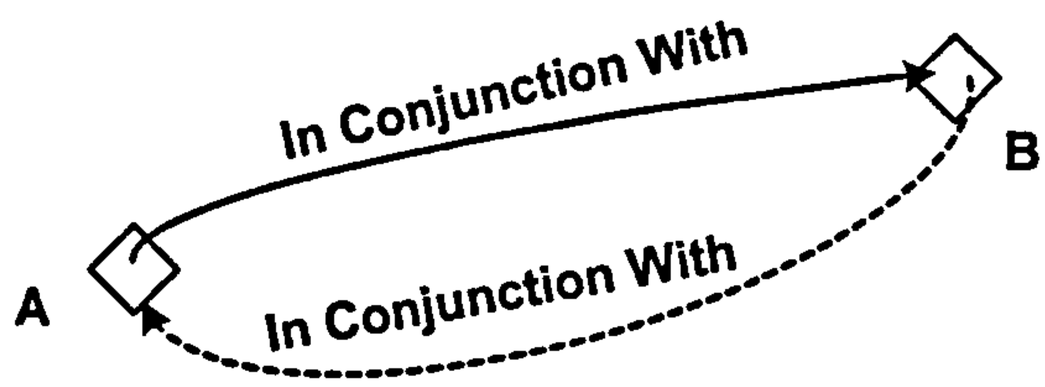


Figure 5-18: An Example of a Symmetric Property: In Conjunction With

Is Used To Generate: describes what can be done with the document and what documents refer to this one. It is an inverse property to the "Refer To".

5.4.3 Annotation of Project Partners

The partners of a project are in the Actor class in the SWIMS Ontology (refer to Section 6.5-Step four-(3)). Each member has a *user profile*. It is a type of metafile annotating attributes/properties of a project partner. Then, SWIMS can provide customised Web Services to them.

(1) User Profile Model

Figure 5-19 illustrates the user profile of a human actor, which is a tabular model with the name, ID, job title, contact details, and interests to name a few.

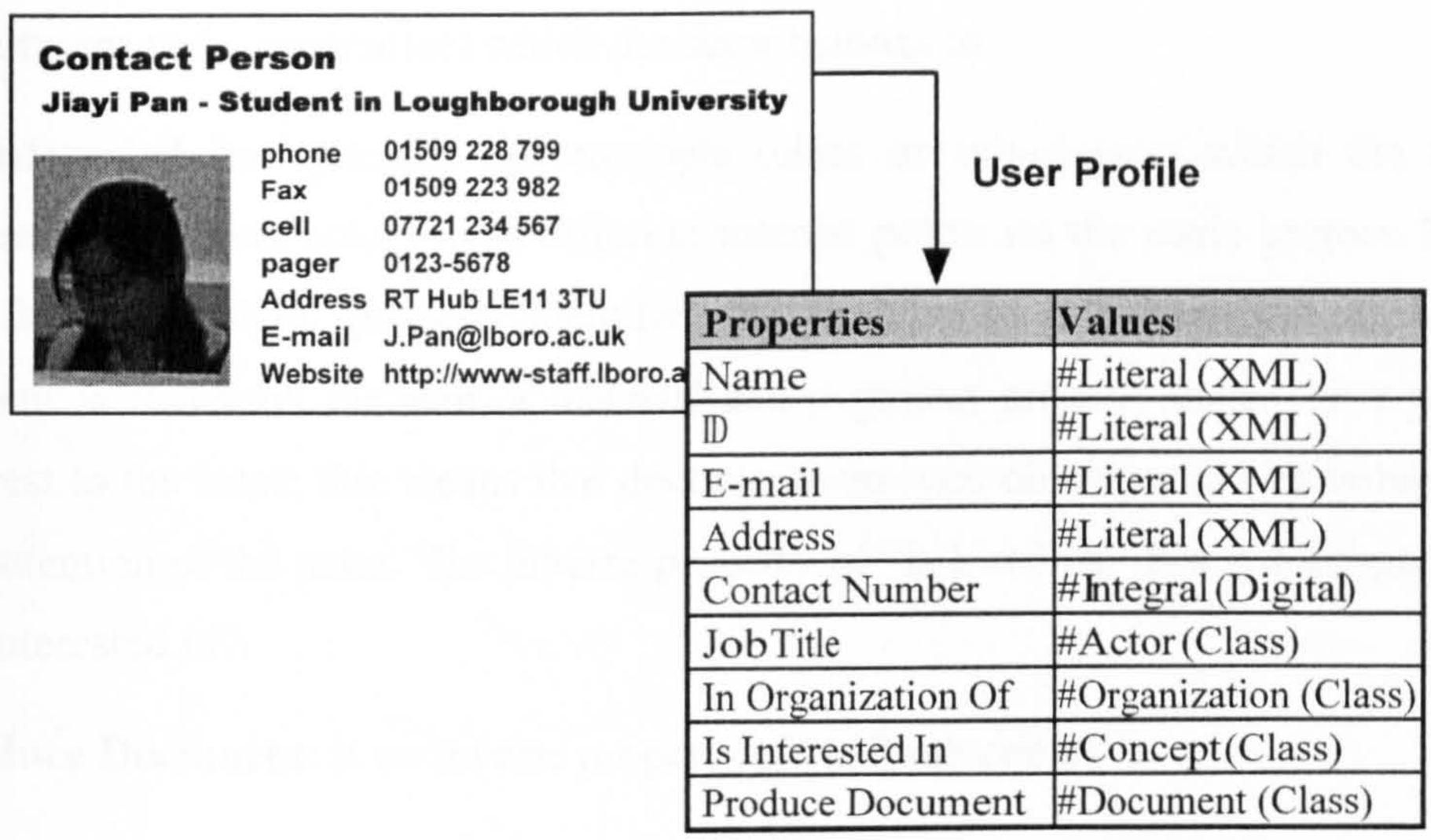


Figure 5-19: User Profile Model

(2) Items of User Profile

In this model, an actor's properties are identified from three considerations: the personal information, contact information and project related information.

(i) "**Personal information**" consists of permanent properties, which will not change in most situations. "Personal information" contains:

Name: the literal expression of the actor's name.

ID: the Uniform Resource Identifier of the actor. This is used to identify the person. Though each individual could have more than one expression of his/her name, he/she can hold only one ID.

(ii)"**Contact information**" contains:

E-mail and **Contact Numbers**: specify the methods to access an actor.

Address: specifies the postal address to access the actor as well as inferring the customised information (e.g. language and local specification system) from the location where the actor is.

(iii) "**Project related information**" contains:

Job Title: the literal expression of the actor's role.

In Organisation Of: specifies the community (a group of actors, e.g. a company, a department and a committee) which the actor belongs to.

Is Interested In: specifies the concepts (class or sub-classes) which the actor concerns. Different actors have different interest points on the same project. Here, the "is interested in" property identifies the concepts (e.g. a document, a subject content, a structural element, a material and a project process stage) that are of interest to the actor; this means that documents on such object would be brought to the attention of the actor. The inverse property of "is Used by" is a sub-property of "is Interested in".

Produce Document: is an inverse property of "is Produced by".

5.5 SUMMARY

This chapter has proposed a framework for construction project information management in the Semantic Web environment. It consists of two major layers and six key components. Better than traditional methods, this approach is compatible with existing Semantic Web technology, which provides content-based Knowledge Management and context-specific Web Services.

As described in the documented project information management scenarios, presented in Section 5.3, SWIMS enables the sharing of vocabulary, resources and relationships by project partners, thus addressing the fragmentation problems (the geographical dispersion and different disciplinary conceptualisations) of construction project team members and facilitating timely collaboration and communication between team members.

To deploy the framework in the documented information management scenarios, the mechanisms and models for annotating project documents and project partners are developed in Section 5.4. In the SWIMS annotation models, entities are annotated by the tabular model with their respective properties. Values of properties could be data, strings, concept classes and instances to classes. The concept classes, properties and relationships between entities are represented by the ontologies.

Documents and project partners are the key entities annotated in SWIMS. Other entities include organisations, structural elements, materials, projects, etc. The

annotation of these entities can be performed in similar models by identifying their properties and values in certain scenarios.

With annotations and ontology, the management of documented information could be more efficient. For example, both computers and human users can establish a document's content by browsing its metafile without reading the document itself and the content can be conversed between different languages and technical systems. The ontology is introduced and explained in Chapter 6 - Ontology Development for SWIMS. Chapter 7 then demonstrates the implementation of the annotations and ontologies in SWIMS; and Chapter 8 introduces the evaluation of the system implementation demonstrator.

CHAPTER 6 - ONTOLOGY DEVELOPMENT FOR SWIMS

6.1 INTRODUCTION

The ontology proposed in this chapter fills in the property slots of the annotation model in Section 5.4. The methodologies for ontology development are reviewed, and the functions and components of the SWIMS Ontology are presented. SWIMS Ontology consists of a concept class hierarchy, a property set and relations between them. The concept class hierarchy and property set are also called taxonomy, which is IFC compliant. Finally the ontology management issue is argued.

6.2 ONTOLOGY DEVELOPMENT METHODOLOGIES

The term Ontology has a long history in philosophy, in which it means the subject of existence. It has been the subject of debate in the Artificial Intelligence (AI) community for more than a decade, and has been defined as the common vocabulary for representing shared knowledge. Ontologies are now ubiquitous in many information management systems from diverse disciplines, such as e-commerce, bioinformatics and medicine. It is the backbone for the Semantic Web (Noy and Musen, 2004). In the Semantic Web environment, ontologies enable computers to access the meaning of information and knowledge, and then process it. Several methodologies are proposed for developing ontologies (Sure, 2002). The most representative ones are reviewed based on three aspects: the steps to develop an ontology, the knowledge domain it is used in, and its supported software.

Cyc methodology (Lenat and Guha, 1990) is used in the Cyc Knowledge Base, a large-scale general knowledge base for all of human consensus reality. The Cyc project has been reviewed in Section 2.3.1-(1). Cyc methodology enables ontology developers use micro-theories to represent different perspectives from various domains on the same knowledge. Also, it allows multiple agents to use the micro-theories. All agents share the core Knowledge Base and contain particular knowledge in their specific domains. This methodology recommends three steps to develop an ontology:

Step One: Manually extracting common sense knowledge;

Step Two: Using machine learning tools to support humans in the codification of new knowledge based on the existing Knowledge Base;

Step Three: Software tools take up the major task of the knowledge acquisition.

The Cyc ontology is extracted from natural language and defined in CycL (Cyc Language). Once the ontology covered enough knowledge, new "common sense" knowledge can be acquired either using natural language or machine learning tools.

OpenCyc is the open source version of the Cyc technology, which has been reviewed in Section 2.4.2-(4) as an ontology editor. Although the technologies for performing the activities of the Cyc methodology have made significant progress since 1990, they did not yet update their methodology, which was proposed in 1990, to meet the development. In this research, Cyc methodology was not adopted because:

1. It is suited to extracting an ontology from human natural language, not the project entities;
2. It uses Cyc Language which is not a widely used Ontology language;
3. This methodology has not been updated for 16 years, but new related techniques have developed extremely fast over the past decade.

Grüninger and Fox's methodology (1994) was adopted by the TOronto Virtual Enterprise project (TOVE, 2006; Fox et al, 1995; Fox and Grüninger, 1998) in modelling the knowledge of business processes and activities. This ontology was developed by answering a set of competency questions and was formally defined in Prolog (a logic programming language). Grüninger and Fox's method recommends building an ontology in six steps:

Step One: Capturing motivating scenarios;

Step Two: Formulating informal competency questions;

Step Three: Specifying the terminology of the ontology within a formal language;

Step Four: Formulating formal competency questions using the terminology of the ontology;

Step Five: Specifying axioms and definitions for the terms in the ontology within the formal language;

Step Six: Establishing conditions for characterizing the completeness of the ontology.

The idea of using competency questions to identify the elements of ontology was adopted in this research. However, the SWIMS ontology was not developed under the Grüninger and Fox's methodology because:

1. The Grüninger and Fox's methodology is too generic to be implemented. No techniques were suggested for performing the activities of the methodology, such as how to formulate the competency questions.

2. It uses Prolog as its formal ontology language while SWIMS does not.

Uschold's methodology (Uschold and Gruninger, 1996) was adopted by the Enterprise project (1997) to codify the terms and definitions relevant to business enterprises in a formal language. The Enterprise ontology was developed, documented and evaluated using the Ontolingua Server, which was reviewed in Section 2.4.2-(2). This methodology provides guidelines for developing an ontology in four steps:

Step One: Identifying purpose;

Step Two: Building the ontology (e.g. ontology capture, coding and integrating existing ontologies);

Step Three: Evaluation;

Step Four: Documentation.

No techniques were suggested for performing the activities of the methodology, such as how to identify the key concepts and relationships in a domain during the ontology capture stage. Therefore, the Uschold's methodology was not the most appropriate methodology adopted in this research.

Bernaras's methodology (Bernaras et al, 1996) was proposed in the Esprit KACTUS project (1996). It is used in building ontologies on the basis of an application knowledge base. This means the ontology development is combined with an application development, especially in the electrical networks domain.

Three ontologies were developed with three applications in the KACTUS project. The applications supported the work of an electrical network, including diagnosing faults, scheduling services resumption after a fault, and electrical network controlling based on the previous two applications. The Bernaras's method recommends building an ontology in three steps:

Step One: Specifying the application;

Step Two: Preliminary design based on relevant top-level ontological categories;

Step Three: Ontology refinement and structuring.

This methodology was not adopted because there was no intention to programme an application with the ontology, and again, no particular techniques were proposed for performing the activities of the methodology.

The Information Science Institute's (ISI) natural language group proposed a methodology in the **SENSUS** project (Swartout et al, 1997), which was used in building an ontology for natural language processing. Ontosaurus is a viewer to browse the SENSUS (a large-scale ontology, which is a framework where additional knowledge can be placed). The SENSUS-based methodology has five steps:

Step One: Identifying a series of terms as seeds in a particular domain;

Step Two: Linking these seed terms manually to SENSUS class hierarchy;

Step Three: Automatically selecting relevant terms in the path from the root of SENSUS to the seed terms;

Step Four: Adding the terms that could be relevant within the domain but have not yet appeared in the path;

Step Five: Adding the entire subtree of some nodes (the node is selected from those that have a large number of paths through them).

Most of these steps require a deep understanding of the knowledge domain to make decisions and are done by hand. No particular techniques were specified for performing the activities of the methodology, such as how to determine whether a node's subtree should be added or not. The SENSUS methodology was not adopted in this research because:

1. It is proposed for extracting ontology for natural language processing, not for projects;
2. It requires too much effort in identifying whether to add the subtrees of a node or not. The decision depends too much on individuals and no particular technique can support the decision making.

METHONTOLOGY (Gómez-Pérez, 1998) was adopted by the CHEMICALS project (Fernández López et al, 1999) and others. METHONTOLOGY is a framework enabling the development of ontologies at the knowledge level. It identified an ontology development process (e.g. evaluation, configuration, conceptualisation, integration, implementation, etc.), proposed a life cycle based on evolving prototypes, and specified the steps and techniques for carrying out each activity. METHONTOLOGY recommends three categories of activities to build an ontology: project management activities, development-oriented activities and support activities.

Project management activities include:

(1) Identifying the tasks to be performed and how they are arranged;

(2) Checking that the tasks were completed as planned;

(3) Ensuring that the quality of the ontology, software and documentation is satisfactory.

Development-oriented activities include:

(1) Specifying the domain and end-user(s) of the ontology;

(2) Structuring the domain knowledge as meaningful models at the knowledge level;

(3) Transforming the conceptual model into a formal or semi-computable model;

(4) Building computable models in a computational language;

(5) Maintaining the ontology.

Support activities include:

(1) Knowledge acquisition;

(2) Evaluation;

(3) Integration;

(4) Documentation;

(5) Configuration management.

METHONTOLOGY is excellent in specifying ontologies at the knowledge level. Different from some other methodologies (e.g. Grüninger and Fox's methodology and Uschold's methodology) using METHONTOLOGY, when developers have a picture of an ontology in their mind, they do not directly code it in a formal language. They express knowledge as a set of intermediate representations (IR) first, and then generate the ontology using a translator software called Ontology Design Environment (ODE) and WebODE (as reviewed in Section 2.4.2-(3)). The use of IR brings significant convenience to developers especially when they are not familiar with the ontology's implementation language. So far, this point was adopted in many ontology editors, including the OntoEdit and Protégé (as reviewed in Section 2.4.2-(1) and (6)). METHONTOLOGY could have been adopted for this research if used in conjunction with WebODE as the ontology development environment. However, in this research, Protégé was considered more appropriate and it has its own ontology development methodology.

A methodology was proposed and applied in the **On-To-Knowledge** (OTK) project, which was reviewed in Section 2.3.1. The recommended ontology development process is illustrated in Figure 6-1 (On-To-Knowledge, 2002) from the feasibility study stage to the evaluation stage, and the maintenance of an ontology was illustrated as the application and evolution stage. This methodology was supported by an OTK Toolset, as depicted in Figure 6-1 and Figure 2-6. This methodology was not adopted in this research for the same reason as METHONTOLOGY while the OTK Toolset was not adopted because it is a large scale ontology development environment which integrated eight ontology related software. An attempt was made to use Ontoshare, a component of the OTK toolset, but there were some configuration problems and difficulties uploading instances to the ontology. Since the focus of this research was not solely ontology development, a framework for the SWIMS Ontology was proposed and some lightweight ontologies were developed

within the framework in the system implementation demos. Protégé is a relatively small ontology development environment but supports most ontology related tasks.

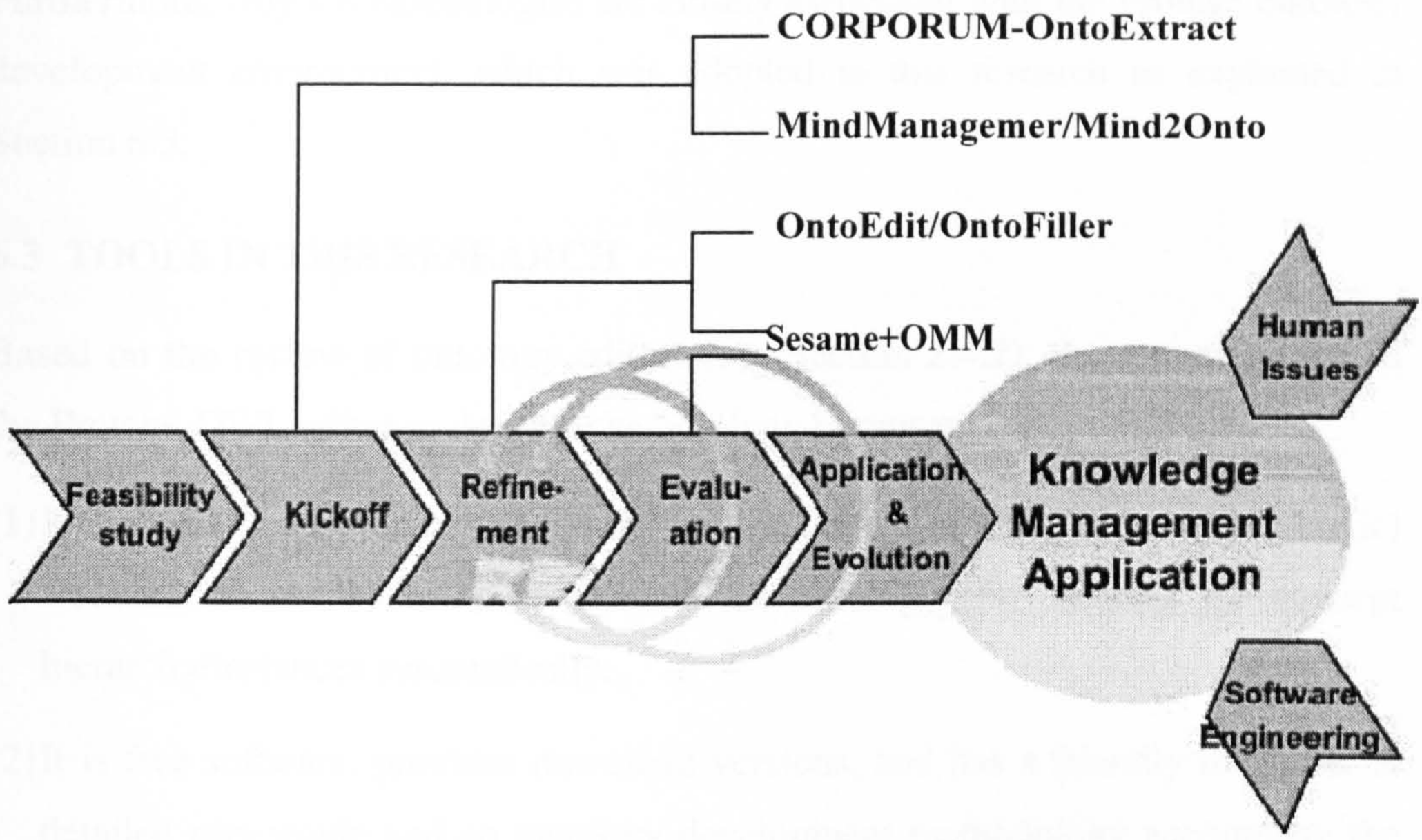


Figure 6-1: OTK Knowledge Management Application Development Process (On-To-Knowledge, 2002)

Referring to Grüninger and Fox's methodology, Uschold's methodology, object-oriented design (OOD) principles and experience from using Protégé-2000 and Ontolingua, **Noy and McGuinness** (2001) proposed a methodology associated with the Protégé ontology editor. They specified seven steps to develop an ontology in an interactive approach. This methodology covered the early stages in an ontology lifecycle (e.g. identification of the domain; consideration of reusing existing ontologies before development; collection of key terms; definition of class, class hierarchy, property and value of the property; and creation of instance to class). Unfortunately, Noy and McGuinness's methodology did not discuss in great detail the ontology management and maintenance issues, unlike METHONTOLOGY. However, Noy and Musen proposed a separate methodology for ontology management associated with PROMPT, a plug-in for Protégé (Noy and Musen, 2001; 2003; 2004). This is discussed in Section 6.8.

Noy's methodologies (Noy and McGuinness's ontology development methodology and Noy and Musen's ontology management methodology) were adopted to construct and manage the first version of SWIMS ontology because they explicitly

specify the fundamental techniques and processes to develop and manage ontologies. They are practical methodologies and especially useful to new ontology developers. Furthermore, Noy's methodologies are closely associated with the Protégé ontology development environment, which was adopted in this research as explained in Section 6.3.

6.3 TOOLS IN THIS RESEARCH

Based on the review of ontology editors (see Section 2.4.2), the research adopted the Protégé-OWL editor to develop the ontology because:

(1) It enables the creation of OWL ontology and the use of a DL (Description Logic) Reasoner to check the consistency of the ontology and compute the concept hierarchy/instances automatically.

(2) It is free software, provides download versions, and has a friendly interface. A detailed user guide and an ontology development methodology accompany the software. Thus, it is easy for a new ontology developer to manipulate it. In fact, it is anticipated that most of the potential ontology developers in SWIMS will be project partners, who are not experts in ontology engineering. They prefer to use tools that are as simple as possible, especially when it would take time to learn the ontology concept itself.

(3) Protégé can be integrated with other widely used ontology-related tools, such as OilEd, under the Collaborative Open Ontology Development Environment (CO-ODE, 2006).

(4) Many plug-ins are available as extensions to the core Protégé software, such as the OWLViz and PROMPT used in this research, and if necessary, it is possible to develop specific software and plug it into the Protégé ontology development environment.

(5) Protégé supports most ontology management activities such as the editing of concepts, properties, instances, rules, consistency checking and ontology merging.

RacerPro was adopted as an OWL-DL reasoner because it is recommended as the "back-end inference system" (reasoner) for use with Protégé. RacerPro was used to

check the consistency of the ontology, classify concepts and infer instances. Meanwhile, FaCT++ was used for double checking the consistency of the ontology.

6.4 FUNCTIONS AND COMPONENTS OF SWIMS ONTOLOGY

6.4.1 Functions

In the Semantic Web environment, as defined in Chapters 2, an ontology is used to specify the knowledge in a specific field, so as to annotate and share information between domain experts. The SWIMS ontology is a specification of a conceptualisation, which defines a common vocabulary for partners who collaborate in a construction project. Therefore, they are able to:

(1) Share common understanding of information among both people and software agents. Partners (and their software agents) use self-developed information management applications. If these applications commit to a commonly agreed ontology, although they use various languages and data structures, they can still communicate on the project issue with each other.

(2) Extend the ontology with new knowledge. A construction project is a complex and dynamic system. The SWIMS ontology architecture allows project partners to integrate new elements (concept, property, relationship and instance) into it or combine other related domain ontologies with it.

(3) Reuse knowledge. A barrier to reusing knowledge is the difficulty of identifying useful information from several records. An Ontology could be used to aggregate information related to a given condition from diverse data repositories. Also, the SWIMS ontology is able to integrate the retrieved information into a specific ontology, if it is not yet in, for completing a certain task.

6.4.2 Components

The SWIMS ontology consists of a formal explicit description of concepts (also called classes) in the construction domain, properties (also called slots or roles) of each concept, and relationships (also called conditions, restrictions or facets) on properties. All of its components are represented by machine-interpretable definitions. Excluded from ontology components, the instances (also called individuals) of each concept also play an important role in managing information within SWIMS. In some literature (Noy and McGuinness, 2001), an ontology and its

associated instances constitute a knowledge base. However, instances and ontology elements (concept, property and relationship) can not be explicitly distinguished because the creation and use of instances are often intertwined with ontology elements. In a well-defined Semantic Web-based system, people can distinguish between the ontology and instances. But in the development process, it is often hard to decide whether an entity is an instance or a subclass of a concept. Both the ontology development issue and instance specification issue are addressed afterwards. In Section 6.5, the SWIMS knowledge base is developed in seven steps with concepts, properties, relationships and instances specified.

6.5 SWIMS ONTOLOGY AND INSTANCES DEFINITIONS

Step One: Determine the domain and scope of the ontology

The domain and scope of the SWIMS Ontology is determined by answering competency questions. The competency question method was initially proposed in Grüninger and Fox's methodology and adopted in Noy and McGuinness's. Here, competency questions are used to find out a sketch of requirements for the SWIMS ontology. Four questions were suggested by Noy and McGuinness:

(1) What is the domain that the ontology will cover?

ANSWER: The SWIMS Ontology will cover the conceptualisation of the construction product and process information. Its first version focuses on the representation of information in project documents.

(2) For what will the ontology be used?

ANSWER: The SWIMS Ontology is being used for capturing and representing the semantics of information in the construction domain, so as to not only manage the information in a single system and single discipline, but also converse, share, search, and retrieve it between heterogeneous resources from multi-disciplinary, geographically distributed project partners. SWIMS Ontology 1.0 is used to test the feasibility of applying the proposed framework (in Section 5.2) to improve the efficiency of project information management between project partners. The scenarios of the SWIMS deployment are described in Section 5.3.

(3) For what types of questions should the information in the ontology provide answers?

ANSWER: The terms that describe a construction project's process, activities, structural elements, materials, documents and actors are important in the SWIMS Ontology. There is no single correct ontology for all scenarios (Noy & McGuinness 2001). If the ontology is used by architects to design the building from scratch, the mapping between functions and structural elements is necessary. If the ontology is used by the structural engineer to process structural design based on the architect's drawings, it would be necessary to map the drawings and specifications to structural elements, specify the connections between structural elements, indicate the materials of the structural elements, and the technical properties of the materials. If the ontology is used by contractors to organise the construction, the cost, duration, and logical and spatial relationships are necessary. If the ontology is used for material procurement, the technical performance and prices of the materials are required. If the ontology is used to exchange information between partners, the synonyms, antonyms, homonyms, mapping between languages, and conversion between terminologies, technical specifications and measurement systems may be needed.

In this research, SWIMS Ontology 1.0 was used to manage documented information for construction projects. It is planned to represent the contents and keywords of project documents, specify the properties of documents, map the dependency between documents, infer concepts by rules, reclassify documents to new categories, and identify the preference of end-users, so as to provide content-based, context-specific and customised services to project partners.

(4) Who will use and maintain the ontology?

ANSWER: As indicated in Section 5.2.5, the SWIMS ontology consists of two parts: the shared ontology and speciality ontologies. The shared ontology specifies the conceptualisation of common core information in the construction domain, which will be used and maintained by all project partners and co-operation agents. A speciality ontology is the specification of information used in a specific task (the management of documented information in this research), which will be used and maintained by domain agents and team members who participate in the task.

Further work, such as enriching the SWIMS Ontology and developing new speciality ontologies, depends on the demands from future research or practical projects.

It is really hard to indicate who will take on the task of managing and maintaining the ontology after it is initially established. In the AI field, the ontology engineer is the professional in charge of the development and maintenance of an ontology. In the e-COGNOS project, a knowledge manager was responsible for the ontology development, mapping to documents, and ontology maintenance (e-COGNOS Consortium, 2002). In SWIMS, every authorized actor can develop an ontology and map it to instances, such as documents. Ontology reasoners will calculate the ontology and check the consistency of concepts automatically. Ontology mergers will compare, analyse and map two ontologies then merge them. However, in some special cases, which require deep understanding of both ontology engineering and civil engineering, the ontology engineer may be called to perform critical ontology-related activities. From the organisational perspective, project partners can either employ their own ontology engineers or share a group of ontology engineers.

Step Two: Consider reusing existing ontologies

Reusing, refining and extending existing ontologies in the construction domain will allow SWIMS to interact with existing applications which are committed to those ontologies or vocabularies. The ontology reuse can be conducted at two levels: reusing a taxonomy/vocabulary/classification system; or reusing a whole ontology. A whole ontology involves a taxonomy (defining the terms of concept, properties and relationships, also called vocabulary) and the use of taxonomy (using the properties and relationships on concepts to represent domain knowledge, also called defining relationships between concepts). Ontologies compliant with the same taxonomy can be easily mapped and merged.

As reviewed in Section 2.3.2, several construction-related applications/software are IFC compliant and many ontology-based applications have adopted IFC model in their taxonomy, such as e-COGNOS construction knowledge management server and APIs (Application Programming Interfaces) (El-Diraby et al, 2003). Other existing construction-related ontologies are fragmented without maturity applications but most of them are also IFC compliant, such as the one for changes in the AEC (Architecture, Engineering and Construction) industry (Cai and Ng, 2004),

collaborative design knowledge ontology (Lai and Carlsen, 2004), a taxonomy for project risk information based on the e-COGNOS ontology (El-Diraby et al, 2006), etc.

Therefore, the SWIMS Ontology was developed to comply with IFC model as well. This enables SWIMS to interact with existing IFC-based applications. As argued in Sections 3.6.1 to Section 3.6.5, the SWIMS taxonomy also adopted Process Protocol model, e-COGNOS ontology and UniClass facets to enrich its IFC-based ontology, such as the concepts about a document's present medium, subject content and dependencies between documents.

Figure 6-2 illustrates the counterparts between the IFC architecture and the SWIMS framework. The SWIMS Shared Ontology is a counterpart of the IFC Kernel and Core Extensions. SWIMS Speciality Ontologies are counterparts of IFC Interoperability layer and Domain modules. SWIMS Documented Information Ontology (hereinafter referred to as "SWIMS Ontology") consists of SWIMS Shared Ontology and two speciality ontologies (SWIMS Product Ontology and SWIMS Documented Ontology). Project partners could have speciality ontology for their own terminology, such as the architect's SWIMS Product Ontology. Ontology management tools can map and merge different ontologies. This issue will be addressed and demonstrated in Sections 6.8, 7.3.1 and 7.3.3-(2)-(ii) and (iii).

Consequently, the SWIMS Ontology is not only IFC compliant, but also capitalised on some other taxonomies. It has a flexible structure, which enables mapping/merging of the SWIMS Ontology to other IFC-based ontologies and interacted with existing IFC-based applications.

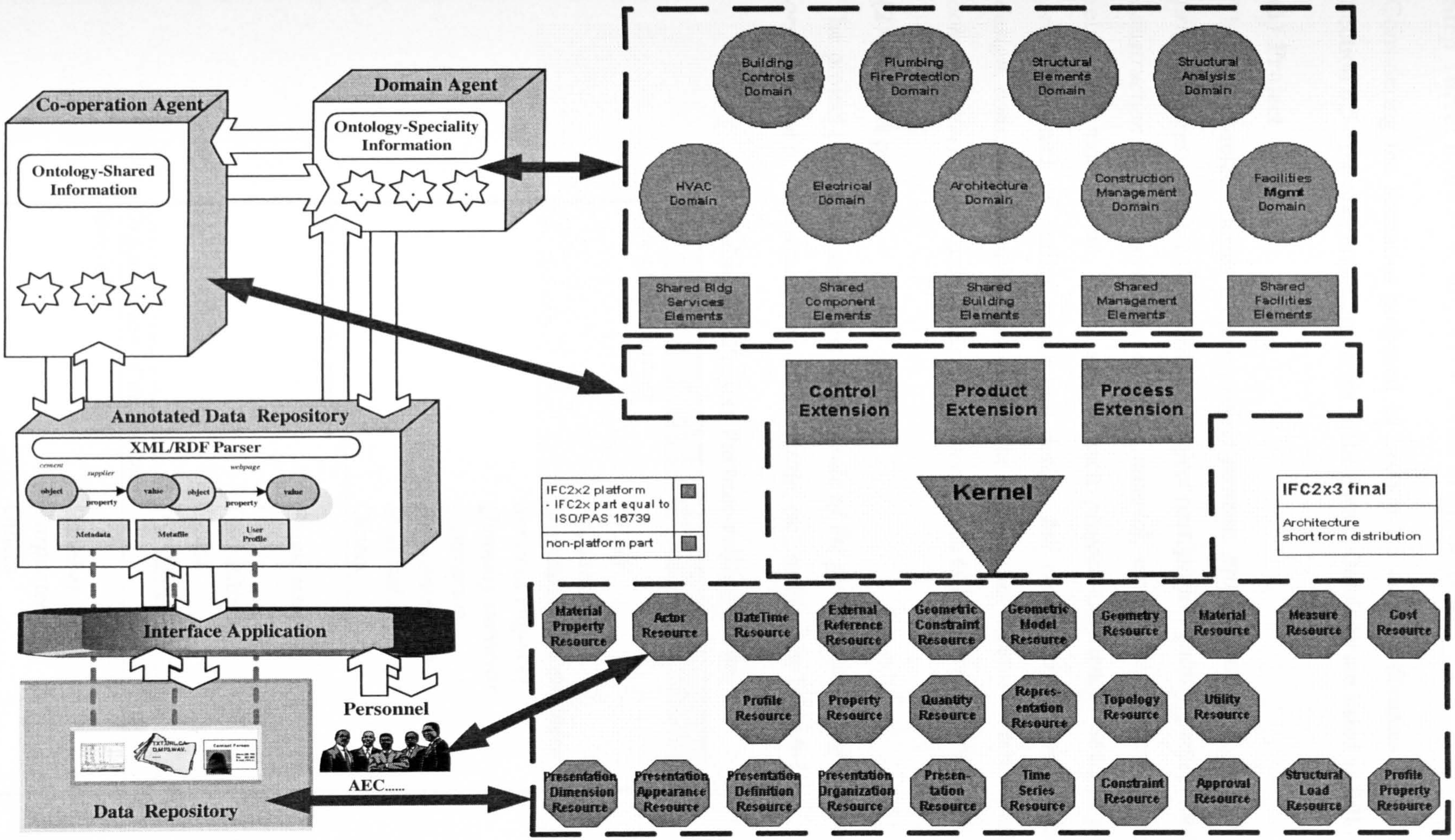


Figure 6-2: The Mapping of SWIMS Framework to IFC Model.

Step Three: Enumerate important terms in the ontology

Considering the scenarios proposed in Section 5.3 and the domains specified in Section 6.5-Step one, important terms in the SWIMS Ontology are listed as follows:

(1) Project

The project-related terms include *project partner*, *project document*, *project type*, *project process*, *project time/duration*, *project cost*, *project quality*, *project location*, *construction site*, *construction activity*, *material*, etc. Some of these terms are related to more terms. For example, each *project* experiences several *project processes*, such as *outline conceptual design*, *full conceptual design*, *production design*, *construction*, *maintenance*, etc. The *project* uses various *materials*, such as *concrete*, *steel*, *stone*, and sub-classes of *stone*, such as *limestone*, *marble*, *rock*, etc.

(2) Project partner

The project partner-related terms include *role of the partner*, *contact of the partner*, *organisation of the partner*, *specialty*, *job title*, etc., as listed in Table 6-1.

Table 6-1: Project Partner-related Terms

Partner	
Role of the partner	
	Client *Architect* *Structural/civil engineer* *Services engineer* *Quantity surveyor* *Contractor* *Sub-contractor* *Material supplier* Others
Contact of the partner	
	Phone number *Fax* *Post address* *E-mail* *Homepage*
Organisation of the partner	
Specialty	
	Plumber *Carpenter* Others

Job title	
	Engineer
	Consultant
	Manager
	Technical staff
	Others

(3) Project type

The project type-related terms include: *building*, *bridge*, *subway*, *airport*, etc.

(4) Building

The building-related terms include: *building type* and *structural element* (as listed in Table 6-2). For example, the *recreational building* is a type of *building* and has sub-classes including *sports centre*, *theatre* and *stadium*. Each *structural element* relates to terms like *materials*, *location*, *cost*, *space*, *shape*, *measurement*, and so on. *Measurement* has the *scale* and *measurement unit*, which may be different from *country* to *country*.

Table 6-2: Building-related Terms

Building			
Building Type			
	Commercial		
	Residential		
	Educational		
	Healthcare		
	Custodial		
	Facility		
	Industrial		
	Recreational		
		Sports centre	
		Theatre	
		Stadium	
	Other		
Structural Element			
	Substructure		
		Foundation system	
		Basement	
	Superstructure		
		Floor	
			Ground Floor
			Upper Floor
		Roof	
		Pillar/column	

		Wall	
			External Wall
			Main Wall
			Partition
		Steelwork	
		Window	
		Door	
	Drainage		
	Stair		
	Services/M&E		
		HVAC/Bldg	*Heating*
			Ventilation
			Cooling
			Air-conditioning System
		Electrical System	
		Water Supply System	
		Gas Supply System	
		Others	
Equipment			
	Elevator		
	Light		

(5) Project document

The document-related terms include *document type*, *subject content*, *path*, *author*, *date*, *format*, *version*, *number*, *subject*, *technical document*, *administrative document*, *architect document*, *legal document*, etc. *Technical documents* should include *drawing* and *specification*. Various types of *specification* exist, including *British code*, *building regulation*, *Act*, and so on. Each *specification* will map to some *structural elements* and *materials*. *Specifications* also linked to *drawings*, including the *drawing*, *figure* and *graph* in the *specification* and the *drawing* refers to the *specification*.

The terms presented above are not an exhaustive set, but comprise a comprehensive list covering items in the documented information management scenario. They will be regarded as elements in the SWIMS Ontology 1.0. The elements (classes, properties and relationships) are defined in the next three steps. Step four, five and six are closely intertwined. The definitions and examples of each top-level SWIMS concept class are specified in Step four, and the definition of SWIMS properties is indicated with a property set in Step five. Step six focuses on refining the definition

of properties by specifying their domains and ranges (value). In some circumstances, Step four (defining concepts) and Step seven (creating instances) should be processed at the same time. This issue will be addressed in Step seven. The identification of relationships goes through the whole process.

Step Four: Define the concepts' classes and arrange the class in a taxonomic hierarchy (subclass- upperclass)

The development of a class hierarchy usually follows one of the following processes: the top-down development process, the bottom-up development process and the combination development process (Noy and McGuinness, 2001; Uschold and Grüninger, 1996).

The top-down method was adopted as the major methodology to define the SWIMS concept hierarchy, because commonly agreed classifications for some SWIMS concepts are available, such as the IFC model and UniClass facets. To certain concept groups, different hierarchy models are coordinated in general, though minor differences exist. Their differences can be treated as synonyms or counterparts in the ontology management stage. In this research, the taxonomies from IFC product model, Process Protocol model, UniClass taxonomy, and e-COGNOS ontology were reused. The hierarchy starts from seven top-level concepts: product, process, actor, control, resource, project and technical topic.

The term object in IFC model, concept in SWIMS and concept in e-COGNUS are counterparts. Hereinafter the concepts are highlighted by underlining.

(1) Product

The class product includes manufactured, supplied or created objects for incorporation into a construction project. The sub-classes of product consist of:

- Physical product, which includes the structural element from the break-down of a project (i.e. door and foundation), temporary element (i.e. formwork), equipment (i.e. excavator), document (i.e. drawing), spatial structure element (i.e. bounding elements), and other elements (i.e. soil and land).
- Non-physical element, which relates to geometric or spatial contexts (i.e. grid, port, annotation, structural actions, and space).

Concepts in this class refer to objects of ifcProduct, ifcElement, ifcBuildingElement and/or terms in UniClass Facet F (space), G (elements for buildings), H (elements for civil engineering works), J (work sections for buildings), L (construction products) and P (materials).

The instances of product are defined by their properties and representations. A typical product instance relates to a geometric or spatial and material context. Sub-classes of product usually hold a shape representation, a local placement within the project structure and one or more material it used.

A product instance occurs at a specific location in space if it has a geometric representation assigned. It can be placed relatively to other products (ultimately relative to the world coordinate system) defined for this project. For example, the relationship *Connect to* is used to capture the structural elements, which are physically connected.

Concepts' *relationship* can be represented in the **properties** associated with the concept, or in the constraints/conditions applied to the concept. The properties can be used to represent and retrieve information related to the concept, while the constraints are used in computing and deducing new relationships between concepts. Hereinafter the properties are highlighted in bold and the relationships are highlighted in italics.

(2) Process

The class process consists of actions taking place in a construction project lifecycle with the intent of designing, costing, acquiring, constructing, or maintaining products or other and similar tasks or procedures. SWIMS Ontology 1.0 is committed to ifcProcess and adopted the Process Protocol model to entail the class. The Process Protocol model divided the lifecycle of a construction project into four broad stages, ten phases, and eight activity zones as reviewed in Section 3.6.1-(1). They are sub-class of process. Please refer to the Process Protocol (2002) project website for the definitions of them.

Each phase and activity zone can have one or more concepts as its sub-class, while a concept can have more than one upper-class. For example, the action to break-down a building into structural elements is a *sub-class* of project management activity

zone, and the task to produce a structural element by a contractor can be a *sub-class* of the construction phase and production management activity zone.

Processes are placed in sequence (including overlapping for parallel tasks) in time. The relationship *sequence* is used to capture the predecessors and successors of a process. For example, demonstrating the need is a *predecessor of* (a sub-class of *sequence*) conception of need.

Processes can have resources assigned to it. This is handled by the relationship *assigned to* the process.

(3) Actor

The actor defines all partners involved in a project during its full lifecycle, such as architect, structural engineer, contractor, project manager, services engineer, quantity surveyor, etc. Each party includes sub-classes people and organisation. The instances of actor provide the support to the human part of resource.

Our definitions of the actor is committed to ifcActor and compliant with UniClass facet B, which defines project partners into nine major subject disciplines, such as architecture, engineering (civil engineering, structural engineering, services engineering and other engineering), surveying (quantity surveying, building surveying, and site surveying), contracting, town and country planning, other construction-related disciplines (such as GIS), other disciplines (such as law and computing).

Every instance of actor is *defined by* an attribute set relating to the actor. The attribute set includes the property information about the actor and relationships that associates the actor to an object. For example, the **ID, owner history, name, contact number** and **description** are information about the actor. The *in organisation of, is acting upon, interest in* and *is assigned to* task are relationships. The relationship where "an actor *is assigned to* a task" is an inverse relationship of "a task *assigned to* an actor". Properties, which are about an actor and used in the documented information management scenario, are listed in the actor annotation model (Section 5.4.3).

(4) Control

The Control is the abstract generalisation of all concepts that control or constrain products or processes. It can be seen as a specification, regulation, cost schedule or other requirement applied to a product or process whose requirements and provisions must be fulfilled.

Each control *is assigned to* product(s), process(es), or other object(s) by using the *assigns to control* relationship. For example, it includes sub-classes, such as space program, cost schedules, order, and work plan.

(5) Resource

The resource defines the information of using something in a process or to produce a product. Things that might be used as resources include the product, actor, and material. For example, the schedule of a project *assigns to* the software MS Project; the construction of the project *assigns to* the actor company A. Here, the MS Project is an instance of both product/software and resource; and certain contractors are instances of both actor/contractor and resource.

(6) Project

The project represents the undertaking of some activities (e.g. design, construction, or maintenance) leading towards a product. The project establishes the context for information to be exchanged or shared. For example the country the *project is located in*, the unit *applied*, the project partners (actors) *involved in*, the *capital invested*, and the *start date*.

(7) Technical topic

The technical topic specifies issues about the attributes, parameters and constraints of the process, product, actor and resources. For example, the project type is a project-related technical issue. Compliance to UniClass facet E (construction entities), the technical topic is enumerated as:

- Construction complexes;
- Pavements and landscaping;
- Tunnels, shafts, cuttings;
- Embankments, retaining walls, etc.;

- Tanks, silos, etc.;
- Bridges, viaducts;
- Towers, superstructures (excluding building);
- Pipelines, ducts, cables and channels;
- Buildings.

Technical issues about construction include:

- Construction phase;
- Construction-related activities (development management, project management, resource management, production management, health & safety, statutory and legal management and process management);
- Construction-related actors (structural engineer, contractor, project manager, supplier and client);
- Construction-related products (specifications, drawings, structural element, equipments, land, soil, material, contract, etc.).

The technical issues about a document include present medium and subject content. The present medium is compliant with UniClass facet A9 "type of medium". The subject content includes project administration, technical, legal, commercial and sub-classes of them (as listed in Section 3.6.1-(3)).

The technical issues about the foundation include foundation types, foundation construction techniques, etc. Technical issues about a raft foundation indicate that the raft foundation is often *used on* soft or loose soils with low bearing capacity to spread the loads over a larger area (normally the entire area of the superstructure), it is usually *used with* crawler excavator, and some related best practices.

The class technical topic collects concepts related to a certain issue, which could be a concept. Probably, these concepts are sub-classes of other concepts. It is no problem of sharing a sub-class between concept classes in the SWIMS Ontology as long as it would not result in an inconsistency.

Step Five: Define the properties of concepts

For answering the competency questions in Step one, the internal structure of identified concepts (from Step four) should be described. Properties and relationships are used for this. Properties are used to define the attributes of each concept by associating concepts with values. A concept inherits properties from its upper-class and can have additional private properties. Relationships specify the linkages between concepts, properties and instances. The development of relationships is intertwined with the development of concepts and properties. The *disjoint* relationship is applied to some concepts if they cannot have any instances in common. For example, document, project_list, actor, structural_element, present medium are disjointed. A property is regarded as a relationship if the property's value is an instance or a concept. Relationships can also be used independently from properties, such as in constraining a concept, specifying rules, and making reasoning.

(1) Type of Property

SWIMS properties fall into three categories: datatype property, object property and annotation property.

- **Datatype properties**

These link the instances of a concept to a XML Schema Datatype value (W3C, 2004 a) or an RDF literal (W3C, 2004 b), such as a string or an integer.

Listing 6-1 lists major datatype property sets in SWIMS Ontology 1.0.

- **Object properties**

These link an instance/concept to an instance/concept. It represents the relationship between two instances/concepts. Some of the relationships represent interdependencies of two instances/concepts. For example, the **SWIMSIsProducedBy** represents an interdependency of a document and an actor and the **SWIMSIsUsedToGenerate** represents an interdependency of two documents. In the speciality ontology, SWIMS Document Ontology 1.0, the interdependencies of documents include: *belongs to* (**SWIMSBelongTo**), *refers to* (**SWIMSReferTo**), *in conjunction with* (**SWIMSInConjunctionWith**), *depends on* (**SWIMSDependOn**), *is quoted in* (**SWIMSIsUsedToGenerate**). The interdependencies of document and

actor include: *use* (**SWIMSIsUsedBy**), *produce* (**SWIMSIsProducedBy**), and *in charge of* (**SWIMSAuthority**).

Listing 6-2 lists major object property sets in SWIMS Ontology 1.0.

Concept/Instance **document**
Properties: ***SwimsDocTitle***
SwimsURI
SwimsIsCharacterizedBy
SwimsUseSoftware
SwimsBriefSummary
SwimsComment
SwimsDateProduced
SwimsDateLastAccessed

Concept/Instance **actor**
Properties: ***SwimsUserName***
SwimsUserID
SwimsUserEmail
SwimsAddress
SwimsContactNumber

Concept/Instance **structural element**
Properties: ***SwimsShape***
SwimsLength
SwimsWidth
SwimsDepth
SwimsWeight
SwimsQualityLevel

Concept/Instance **material**
Properties: ***SwimsQuantity***
SwimsPrice
SwimsShape
SwimsLength
SwimsWidth
SwimsDepth
SwimsWeight
SwimsQualityLevel

Concept/Instance **project**
Properties: ***SwimsBudget***
SwimsStartDate
SwimsEndDate
SwimsDuration
SwimsCost
SwimsHumanDay

Listing 6-1: Dataype Property Set on SWIMS Ontology

Domain	Property	Range
document	*SwimsInProjectOf*	project list
	SwimsInCategoryOf	document type
	SwimsIsAbout	material structural element activity zone
	SwimsInFormatOf	present medium
	SwimsAuthority	actor organisation
	SwimsIsProducedBy	actor organisation
	SwimsIsUsedBy	actor organisation
	SwimsLastAccessedBy	actor organisation
	SwimsVersion	version
	SwimsReferTo	document
	SwimsIncludeDocument	document
	SwimsBelongTo	document
	SwimsIsUsedToGenerate	document
	SwimsDependOn	document
	SwimsIsAppliedIn	activity zone process stage
	SwimsInConjunctionWith	document
actor	*SwimsInOrganizationOf*	Organisation
	SwimsIsInterestedIn	concept
	SwimsProduceDocument	document
	SwimsCooperateWith	actor
	SwimsIsAssignedTo	structural element
structural element	*SwimsConnectedTo*	structural element
	SwimsIsPartOf	structural element
	SwimsConsistOf	structural element
	SwimsInProjectOf	project
	SwimsUseMaterial	material
	SwimsAssignTo	actor
material	*SwimsIsUsedToMake*	structural element
	SwimsIsProducedBy	material supplier
process	*SwimsUseDocument*	document
	SwimsInvolve	actor

Listing 6-2: Object Property Set on SWIMS Ontology

- **Annotation properties**

They are used to add information (metadata) to concepts, instances and datatype/object properties. Figures 6-3 to 6-6 are examples of the annotation properties for a concept, an instance, a datatype property and an object property. The annotation property is the shaded slot on the right hand side of each window.

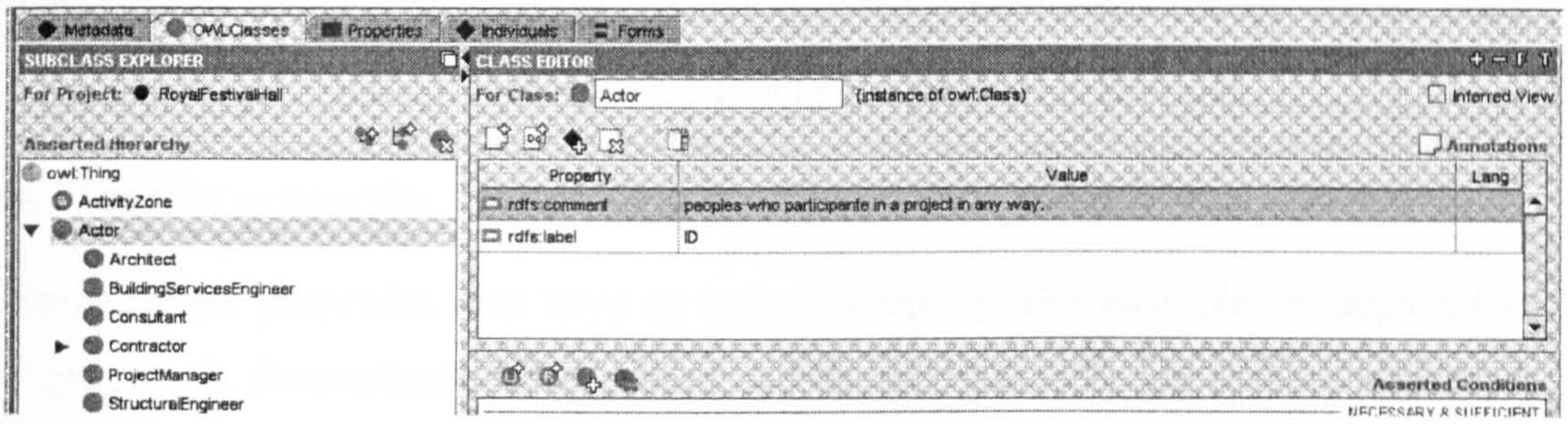


Figure 6-3: Annotation Property of a Concept - actor

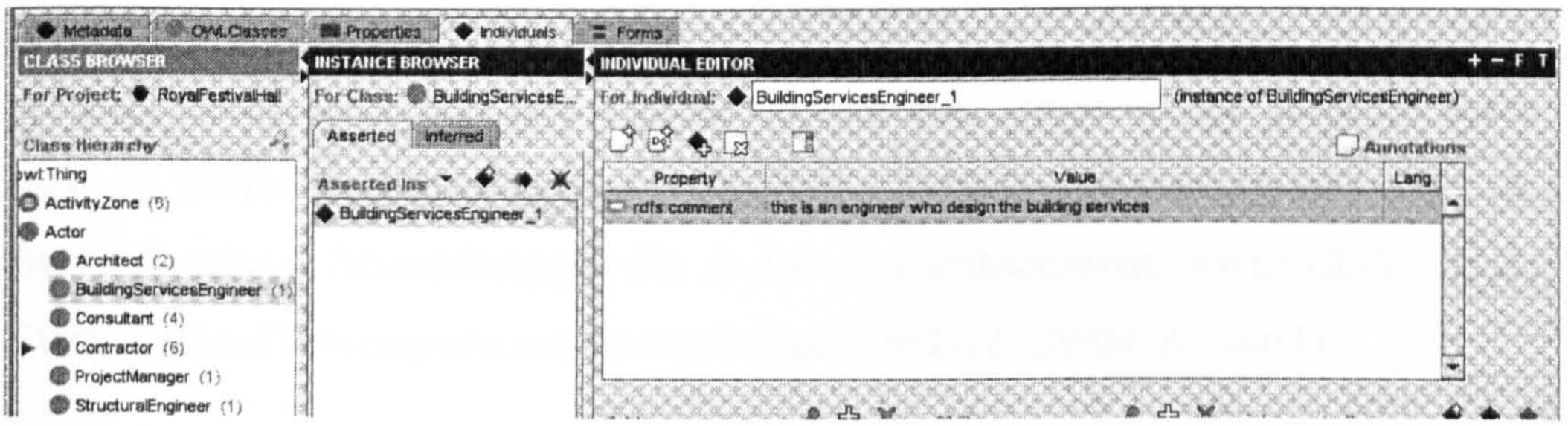


Figure 6-4: Annotation Property of an Instance - a building service engineer

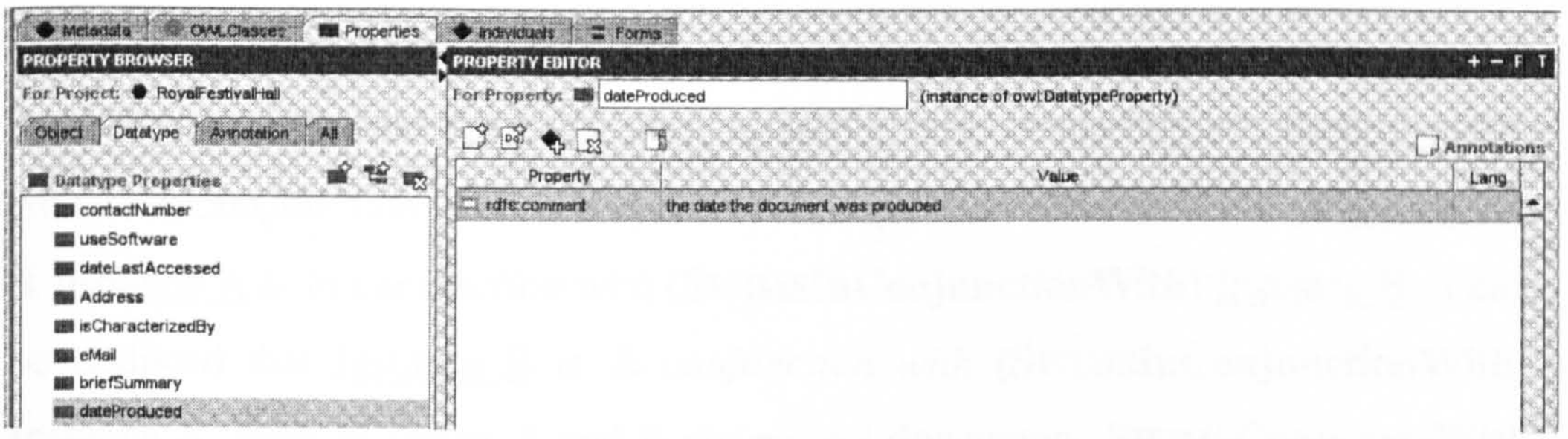


Figure 6-5: Annotation Property of a Datatype Property - *DateProduced*

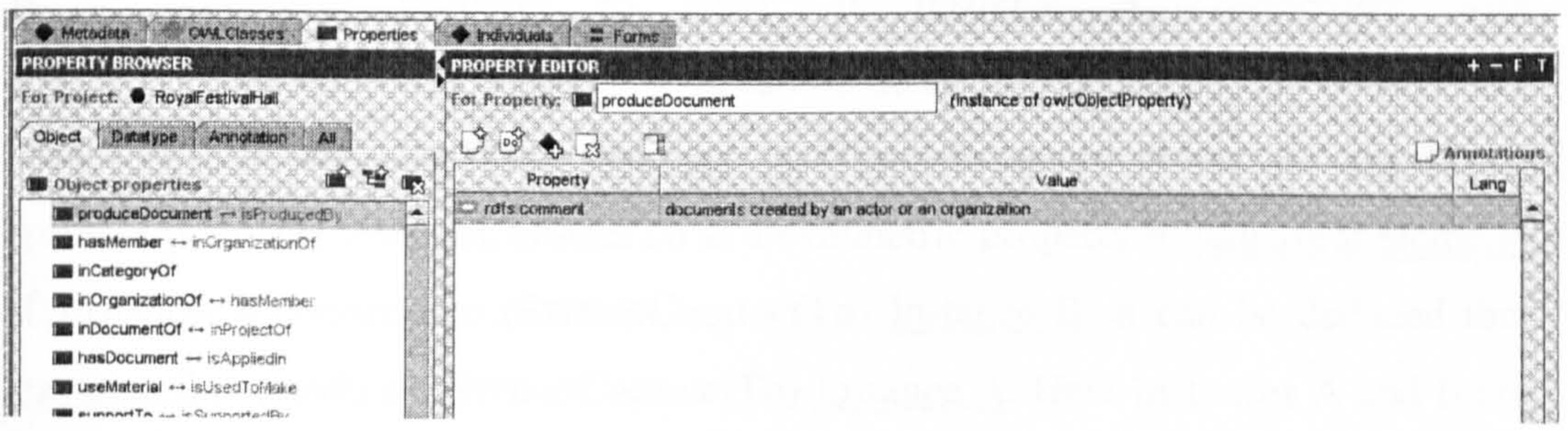


Figure 6-6: Annotation Property of an Object Property - *ProduceDocument*

(2) Property Characteristics

- **Functional properties**

An object property is a functional property if only one instance can be related to a given instance via the property. That means the property allows at most one instance to be its value. For example, as depicted in Figure 5-15, **SWIMSLastAccessedBy** is a functional property to instance/concept document and actor. The properties **URI**, **SWIMSUserID**, **SWIMSUserEmail** are functional properties as well.

- **Inverse properties**

Some object properties may have an inverse property. For example, as depicted in Figure 5-17, **SWIMSIncludeDocument** and **SWIMSBelongTo** are inverse properties to instance/concept document. If Instance A *includes* (**SWIMSIncludeDocument**) Instance B, it can be deduced that Instance B *belongs to* (**SWIMSBelongTo**) Instance A. Here Instances A and B are project documents. Other inverse property pairs include: **SWIMSUseMaterial** & **SWIMSIsUsedToMake**, **SWIMSIsProducedBy** & **SWIMSProduce**, **SWIMSIsAppliedIn** & **SWIMSUseDocument**, **SWIMSReferTo** & **SWIMSIsUsedToGenerate** and **SWIMSIsAssignedTo** & **SWIMSAssignTo**.

- **Symmetric properties**

Some object properties may be a symmetric property if the two instances/concepts it linked can be interchanged. For example, as depicted in Figure 5-18, **SWIMSInConjunctionWith** is a symmetric property to instance/concept document. If Instance A *is in conjunction with* (**SWIMSInConjunctionWith**) Instance B, it can be deduced that Instance B *is in conjunction with* (**SWIMSInConjunctionWith**) Instance A. Here Instances A and B are project documents. **SWIMSCooperateWith** is a symmetric property to actor. If Instance A *co-operates with* (**SWIMSCooperateWith**) Instance B, it can be deduced that Instance B *co-operates with* (**SWIMSCooperateWith**) Instance A. Here Instances A and B are actors (project partners). **SWIMSConnectTo** is a symmetric property to structural element. If Instance A *connects to* (**SWIMSConnectTo**) Instance B, it can be deduced that Instance B *connects to* (**SWIMSConnectTo**) Instance A. Here Instances A and B are structural elements.

- **Transitive properties**

If an object property is transitive, and the property relates instance/concept A to instance/concept B, and also instance/concept B to instance/concept C, then it can be inferred that instance/concept A is related to instance/concept C via the property. For example, as depicted in Figure 5-16, **SWIMSReferTo** is a transitive property to instance/concept document. If Instance A *refers to* (**SWIMSReferTo**) Instance B, and Instance B *refers to* (**SWIMSReferTo**) Instance C, it can be deduced that Instance A *refers to* (**SWIMSReferTo**) Instance C. Here Instances A, B and C are project documents. Other transitive properties to document include: **SWIMSBelongTo, SWIMSIsUsedToGenerate, SWIMSConsistOf,** and **SWIMSIsPartOf.**

Step Six: Define the allowed values for properties

A property can have different facets describing its value type. Some facets of properties have been decided in the previous step with the definition of properties. This step focuses on checking the allowed value types, the number of values (cardinality), the domain and ranges of a property.

(1) Value Type

To datatype properties, the type of their value could be explicitly specified. For example, the **SWIMSDocTitle, SWIMSCharacterisedBy, SWIMSBriefSummary, SWIMSComment** and **SWIMSUserName** have String value. The **SWIMSDateProduced, SWIMSDateLastAccessed, SWIMSContectNumber, Length, SWIMSWidth, SWIMSDepth, SWIMSWeight, SWIMSQuantity, SWIMSPrice, SWIMSCost, SWIMSBudget** and **SWIMSHumanDay, SWIMSDuration** have Number value (including Float and Integer as more specific value types if necessary). The value of **SWIMSDateProduced, SWIMSDateLastAccessed, SWIMSStartDate, SWIMSEndDate** are in Date type. **SWIMSShape** and **SWIMSQualityLevel** have enumerated values. The Boolean type of value is not used in the SWIMS Ontology 1.0, but could be used in the future.

(2) Cardinality

The parameter cardinality defines how many values a property slot can have. For example, the property **SWIMSDateProduced** can only have a value and **SWIMSIsUsedBy** can have multiple values.

(3) Domain and range of a property

This issue has been addressed in Listing 6-2. The range of a property slot indicates the allowed classes of instances for this property. For example, the range of **SWIMSInProjectOf** allows instances from the project list. The domain of a property indicates the classes, to which this property is attached, or which is described by the property. For example, the document and structural element are the domains of **SWIMSInProjectOf.**

Properties can be regarded as constraints to domain concepts and used in defining new concepts or retrieving instances. It is neither necessary nor possible to identify all the possible information about the domain. SWIMS Ontology 1.0 does not cover all the possible properties of concepts in construction domain. It only covers those that will be used in the implementation of SWIMS for documented information management, which is demonstrated in Chapter 7.

Step Seven: Create instances to fill in the value for properties

After the previous six steps, a simple ontology has been developed. Figure 6-7 is a screenshot of the concept document (on the left hand side of the window) and its properties (the metafile, which is on the right hand side of the window) in Protégé. Then, instances from the real world, such as project documents and partners, can be linked to the ontology to enable computers to access the meaning of them. The creation of instance involves three steps:

(1) Choosing a concept;

(2) Creating an individual instance of that concept;

(3) Filling in the property values with real-world information.

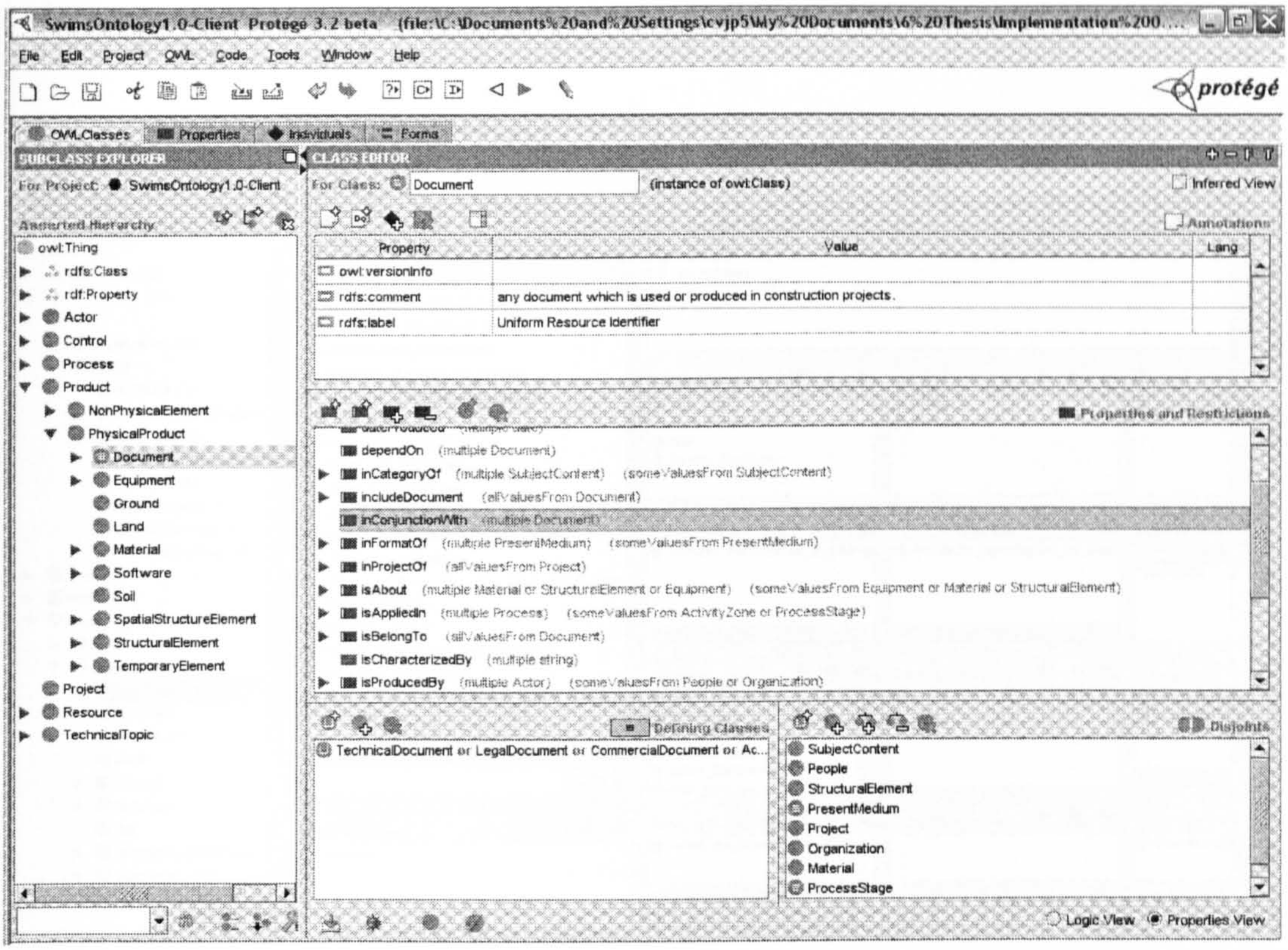


Figure 6-7: Screenshot of Document Class in Protégé

Figure 6-8 shows an instance of document about the Royal Festival Hall Extension Building project. The instances of document are listed in the middle of the window and the metafile of the selected instance is displayed on the right hand side of the window. Figures 6-9 and 6-10 show the properties and instances of organisation.

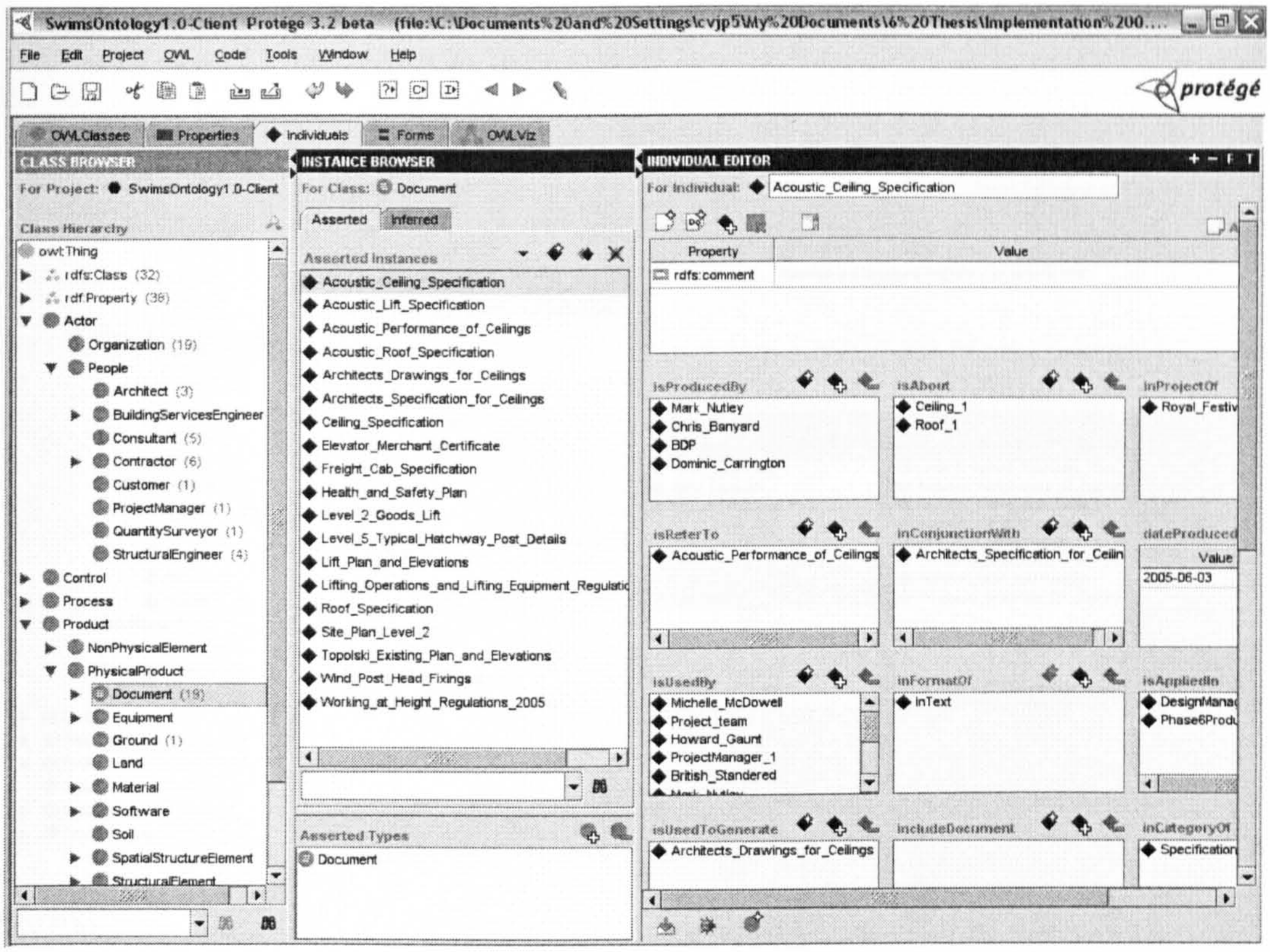


Figure 6-8: Screenshot of a Metafile for a Project Document in Protégé

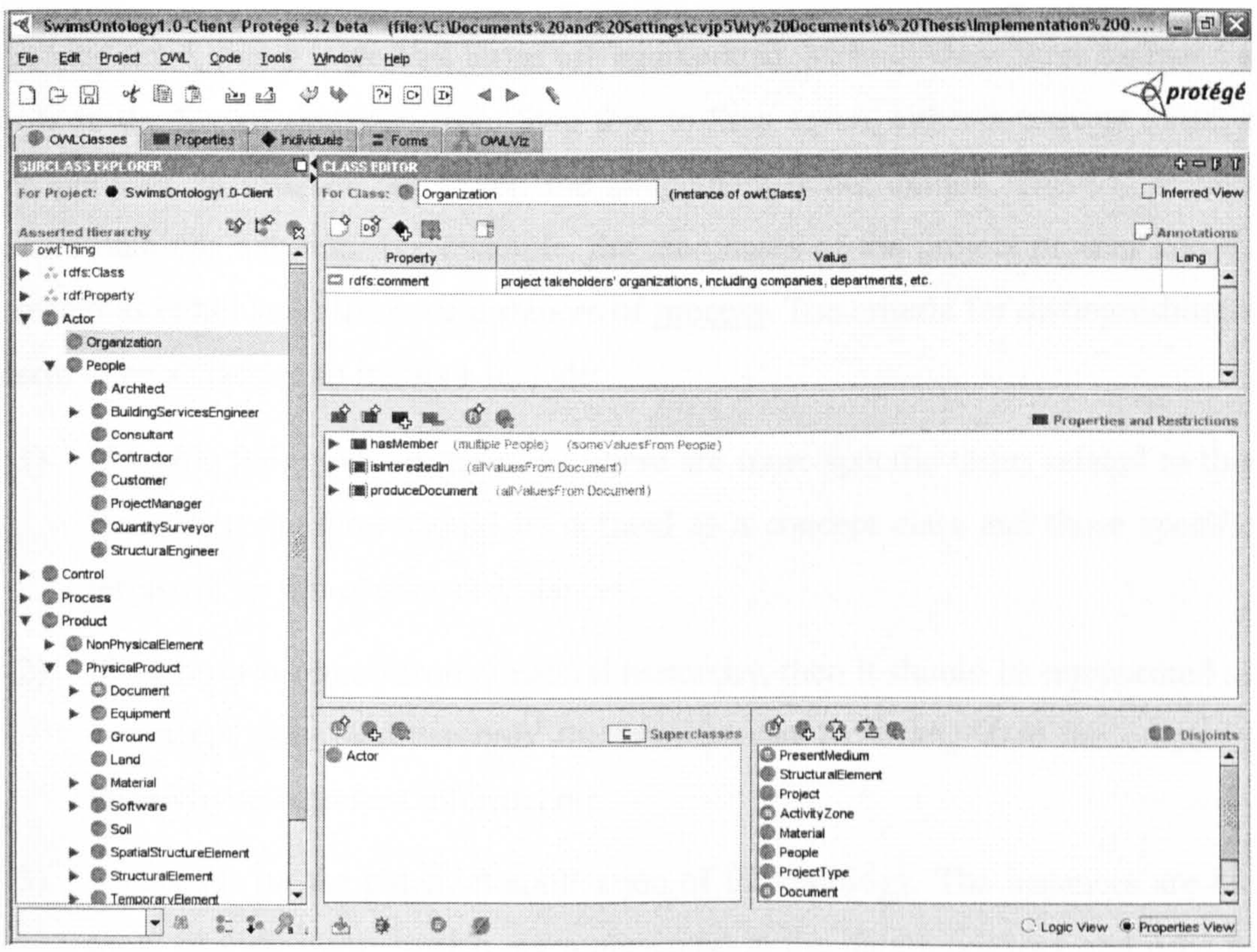


Figure 6-9: Properties of an Organisation Actor

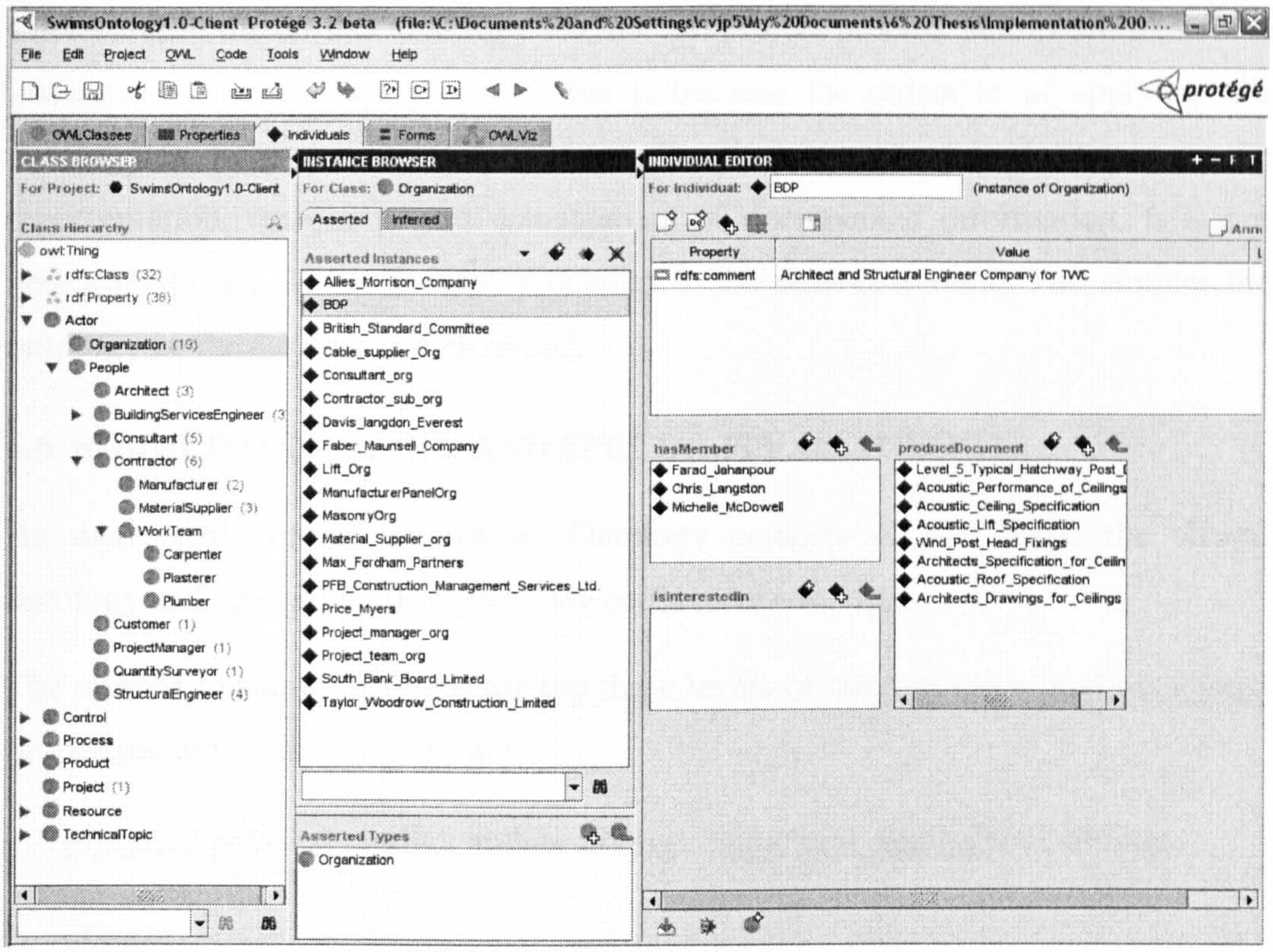


Figure 6-10: Screenshot of a Metafile for an Organisation in Protégé

In Step three, many important terms are enumerated. Most of these were assigned a role in the SWIMS ontology from Step four to Step seven, either a concept class, a property or an instance. However, the assignment is not unique. There is always more than one solution. For example, the ten phases of the project process can be treated as either sub-classes or instances of process. The criteria for distinguishing a term to be a class or an instance include:

(1) If within the important term list, there are more specific terms related to this term, then the term should be defined as a concept class and those specific terms are its sub-classes or instances;

(2) If the term is reused from a natural hierarchy, then it should be represented as a concept class because only the concept can be arranged in the ontology hierarchy to represent information;

(3) It depends on the potential application of the ontology. The instances are the most specific items, which are represented in the data repository and used to answer the competency questions identified in Step one.

Therefore, in the SWIMS Ontology 1.0, the ten project process phases are treated as instances of the concept process. That is because the objective of applying this ontology is to generate demonstrations for showing the use of SWIMS in the representation, reasoning, and management of documented information. It is not necessary to list all the detailed processes in the demonstrations. The simpler the ontology is, the easier it is understood.

6.6 SHARED ONTOLOGY AND SPECIALITY ONTOLOGY

As mentioned before, the SWIMS Ontology consists of two parts: the shared ontology and speciality ontology. They could have overlaps.

The **shared ontology** includes the top three levels of concept class (and associated properties and relationships), such as:

- Product / physical product and its siblings / structural element and siblings;
- Process;
- Actor/ architect and its siblings;
- Control;
- Resource/ material / steelwork and siblings;
- Project;
- Technical topic.

The **speciality ontology** includes concepts related to a specific domain. There are two speciality ontologies in this research: the product ontology and the document ontology. The product ontology includes detailed concepts (and associated properties and relationships) related to product concept (especially building sub-concept in this research), such as the sub-concepts of structural element and equipment. The document ontology includes concepts (and associated properties and relationships) related to document concept, such as document type, format, software, some related actors, resources, structural elements and equipments. The properties of document in the speciality ontology are identified by answering four questions, as specified in Section 5.4.2-(2):

- What is it about (Content of Document)?

- Who is it meant for?
- What supporting documentation or analysis has been used?
- What can be done with the information?

The value slots of these properties are also defined according to the four questions. The details of SWIMS Shared Ontology 1.0 and speciality ontologies refer to Sections 7.3.1-(1) and (2).

SWIMS Ontology 1.0 is a lightweight ontology with about 200 concepts. Please refer to Appendixes b, c and d for the concept hierarchy of each ontology. In this research, the speciality ontologies were mapped and merged to the shared one using PROMPT in the Protégé ontology development environment, and performed some further operations as an overall ontology within the implementation scenario. Demos are presented in Chapter 7 to demonstrate the implementation of SWIMS Ontology 1.0 in managing documented project information.

6.7 ONTOLOGY VALIDATION

The taxonomy (the terms used in naming concepts/properties/relationships and the class hierarchy of concepts/properties) of SWIMS Ontology 1.0 is derived from the IFC model, Process Protocol Model, UniClass taxonomy and e-COGNOS ontology (which also committed to the IFC model and UniClass taxonomy). The rationality of the taxonomy of SWIMS Ontology 1.0 is inherited from its origins. Therefore, the validation of the ontology focused on the consistency checking for the relationships/links between the taxonomy elements (concepts/properties/relationships). As reviewed in Section 2.4.3, there are two leading OWL-DL reasoners, FaCT++ and Racer. Because the RacerPro is the reasoner used in the implementation demos, FaCT++ was adopted here to double check the consistency of SWIMS Ontology 1.0. The consistency of class and instances, value of each property and the relationships between concepts and properties were calculated and inconsistencies of some property values were found. After corrections, the accuracy of SWIMS Ontology 1.0 was examined and the elements in it are consistent.

6.8 ONTOLOGY MANAGEMENT

Ontology management covers various issues in the use of existing ontologies. For example, it includes ontology storage and retrieval, ontology evolution and versioning, reuse and combined use of ontologies.

SWIMS Ontology 1.0 is the first version of SWIMS Ontology. It is lightweight and has only been used in the implementation demos for this research. Therefore, the evolution issue is vital when reusing and extending it for other applications in the future. This is why the SWIMS ontology was divided into two parts. The most generic concepts are defined in the shared SWIMS ontology, and the domain-related concepts are defined in the speciality SWIMS ontologies. These two types of ontologies constitute an ontology library. This architecture allows self-developed ontologies to be developed, stored, connected and executed in the common SWIMS framework. For example, the extension of the shared ontology can be performed at any time, while a speciality ontology will be developed when applications will use information from a specialty domain. The changes of one ontology will not affect elements of other ontologies but will lead to changes in the merged ontology of this one with other ontologies. In short, every ontologies can be developed separately, although they should commit to the IFC model.

Another important issue in this research was the ability of mapping/merging/integrating the SWIMS Ontology 1.0 to other construction domain ontologies. To ensure the portability, SWIMS Ontology 1.0 was based on the IFC model, the most widely-accepted conceptualisation model for construction products. Therefore, it can map to other IFC-based ontologies under the common IFC taxonomy and interact with those existing IFC-based applications.

The combination of the SWIMS Ontology with other ontologies plays an important role in reusing the SWIMS Ontology. **Ontology combination**: means the use of different ontologies together for a task, in which, their mutual relation is relevant. This operation consists of several tasks working on the ontologies (de Bruijn et al, 2006):

Ontology mapping: relates similar concepts or relations from different sources to each other by the *same as* relation. This results in a virtual integration.

Ontology alignment: means the commitment of different ontologies to a mutual agreement, making them consistent and coherent.

Ontology merger and **integration:** means creating a new ontology from diverse existing ontologies, which have overlapping parts. Two types of merger exist: virtual merger and physical merger. The most significant difference between the two mergers is a new ontology which will physically exist after the physical merger, but not after the virtual merger.

Ontology translation or transformation: means changing the representation formalism only or slightly changing the semantics of an ontology.

These tasks are interrelated and have common elements and subtasks. **Aligning** and **merging** ontologies are the central aspects of ontology combination. The combined use of multiple ontologies often start from **mapping** concepts from different ontologies, and then aligning the separate ontologies or merging/integrating them into a new ontology.

The method of combining different ontologies includes four steps (McGuinness et al, 2000 and Noy and Musen, 2003):

Step One: Finding the similarities between overlapping ontologies;

Step Two: Comparing and analysing elements (concepts or properties) and relations;

Step Three: Relating the ones that are semantically close via the *same as* and subsumption relations;

Step Four: Checking the consistency, coherency and non-redundancy of the result.

Several problems may occur during the ontology mapping and merging, such as mismatches from language level or ontology level, name conflicts, dangling references, redundancy in the class hierarchy and inconsistencies. Thereby, ontology combination is a complicated process, which requires a deep understanding of both ontology engineering and domain knowledge. Though, this task is mostly done manually by the ontology engineers, some software tools are being developed for facilitating it, such as PROMPT (Noy and Musen, 2003). As a

set of plug-ins of Protégé, PROMPT suite was adopted in this research as an ontology management tool. It can determine the conflicts in the merged ontology for performing an operation, propose possible solutions to the conflict, consider the ontological elements related to the conflict, and propose other operations that should be performed. The SWIMS Shared Ontology and Speciality Ontoloiges were mapped and merged using PROMPT. PROMPT automatically examined the name conflicts and dangling references in ontologies and proposed solutions to conflicts. The mismatches between different ontology elements were avoided by selecting the most significantly same concepts to map/merge first and then identifying the other concepts based on those mapped/merged concepts. Finally, the redundancy in the class hierarchy and the consistency of merged ontology was checked by ontology reasoner (RacerPro). Please refer to Chapter 7 for the demonstration of ontology mapping/merging in this research.

The PROMPT suite consists of three components, as depicted in Figure 6-11 (Noy and Musen, 2004). The component PromptFactor was deleted in the update of PROMPT framework from the 2003 version (Noy and Musen, 2003) to 2004 version (Noy and Musen, 2004). PromptFactor is a tool that enables users to factor out part of their ontology into a new sub-ontology. Here the 2004 version was adopted as the factor function was not considered necessary.

iPrompt is an interactive ontology-merging tool, which assists users in merging ontologies by providing suggestions, analyzing conflicts, and suggesting conflict-resolution strategies.

AnchorPrompt (Noy and Musen, 2001) is a graph-based tool for finding related concepts in different ontologies.

PromptDiff (Noy and Musen, 2004) is an ontology-versioning tools that determines what has changed between two versions.

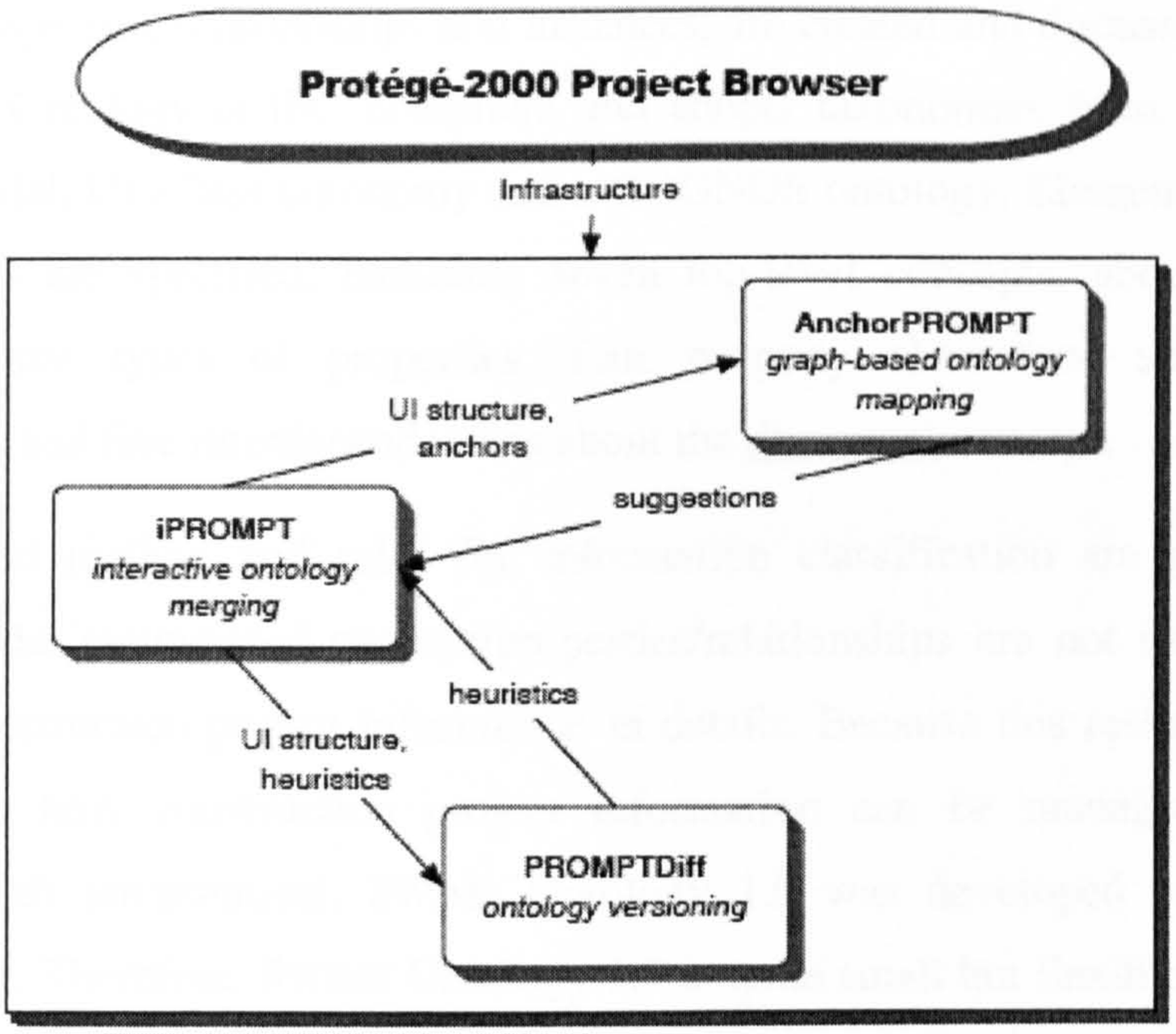


Figure 6-11: Prompt Ontology Management Framework (Noy and Musen, 2004)

These components closely co-operate with one another. iPrompt provides a user-interface for other tools. This component allows users to browse the two ontologies side-by-side, highlight concepts from different ontologies by colors, and list related terms in pairs. The related term pairs are provided to AnchorPrompt and analyzed in AnchorPrompt. Then the analysis is returned to iPrompt and additional suggestions can be presented to the user. PromptDiff uses some of the heuristics from iPrompt in the comparison of ontology version.

All the components can be plugged into the Protégé ontology development environment. They were used to merge SWIMS speciality ontologies and the shared ontology, and observe different versions. These operations are demonstrated in Chapter 7.

6.9 SUMMARY

The SWIMS Ontology was developed to represent the information in construction projects, especially the scenarios of documented project information management proposed in Chapter 5. The development of SWIMS ontology adopts Noy and McGuinness's methodology, which consists of seven steps. Elements of the SWIMS Ontology, which are called the knowledge base by Noy and McGuinness, including

concepts, properties, relationships and instances, are created and discussed in detail. The SWIMS Ontology is IFC compliant and adopts taxonomies from the Process Protocol model, UniClass taxonomy and e-COGNOS ontology. Elements of SWIMS ontology 1.0 are specified, including seven top-level concepts, about 200 sub-concepts, three types of properties, four property characters, six types of relationships and five interdependencies about the document concept.

The proposed method and rules for information classification are generic and correct but the enumerated concept/properties/relationships are not exhaustive to cover all construction project information in details. Because this research aims at investigating how construction project information can be managed within a Semantic Web environment, SWIMS Ontology 1.0 was developed only for the demonstrator. Therefore, SWIMS Ontology 1.0 is quite small but flexible, extensible and compliant with most existing taxonomies/classification systems about construction/building works and interacts to all IFC-committed applications. SWIMS Ontology 1.0 would doubtlessly need extending and could be extended in the future to satisfy the complex heterogeneous construction environment.

OWL-DL is used to edit the SWIMS Ontology in the research. Protégé-OWL and RacerPro were adopted as the ontology editor and reasoner. Some ontology management-related issues were discussed in general and PROMPT, an ontology merging and aligning software tool suite, is reviewed in particular. Since ontology management is a complicated task and involves many sub-tasks, in this research, ontology mapping and merger are executed as being representative of other ontology management tasks. Ontology management examples are presented in the Semantic Web-based documented information management application in the next Chapter.

SWIMS Ontology 1.0 was applied in the metafiles of documents, user profiles of project partners and metadata of structural element's data. Therefore, computers can perform content-based document management activities, such as customising the document classification automatically, keeping information consistent throughout diverse resources, searching for documents based on its semantics, converting different terminologies, etc. The implementation demos are displayed in the next Chapter.

CHAPTER 7 - IMPLEMENTATION OF SWIMS DEMONSTRATOR

7.1 INTRODUCTION

The SWIMS documented information management application is presented in this chapter. The functions of the application are illustrated using IDEF0 diagrams. The deployment of SWIMS is introduced from two perspectives: the "to-be" status is described from the user's perspective in scenarios in Section 5.3 and the "as-is" status is explained from the ontology engineer's perspective in the following sections. This chapter focuses on how to process SWIMS documented information management tasks rather than the user-interface design. To demonstrate the application, several examples are provided to explicitly display the techniques behind the end-user interface from the ontology engineer's perspective. An end-user interface for a document search is provided towards the end of the chapter.

7.2 IMPLEMENTATION ENVIRONMENT

The implementation environment includes both the construction actors and IT tools. Actors are all partners involved in the construction project, which include both people and organisations. IT tools include both hardware and software systems.

7.2.1 Hardware

(1) From the End-user's Perspective:

All project partners are the end-users of SWIMS. The use of SWIMS does not require extra expense on hardware other than using traditional Web-based information management systems. Desktops, laptops, cell phones and PDAs can all be used as terminals to access information on SWIMS through a network (Aziz et al 2004).

(2) From the Ontology Engineer's Perspective:

One or more computer servers are needed to store and handle SWIMS ontologies. The annotated information (metafiles) is stored with original information (electronic files), so the hardware for storing metafiles is shared with the one storing original information.

The hardware requirements for running most Semantic Web tools generally include: a Pentium or equivalent running Windows 2000, Windows NT, Athena, or Linux, with at least 512 megabytes of RAM. The performance of the application depends on the scale of the ontology and information repository. It is also proportional to the speed of the processor. Demos in this research are programmed on a Toshiba A10 laptop with Microsoft Windows XP operating system. Its main technical features are: 2.50 GHz processor, 752 megabytes of RAM, 40 GB hard disk and other facilities. The final application of SWIMS can be deployed on a group of computers which are connected through a network.

7.2.2 Software

(1) From the End-user's Perspective:

End users are able to access SWIMS through an Internet Explorer (or similar) browser. Because the Semantic Web changes the way in which information is processed, and the information management is ubiquitous, there is no key application for the Semantic Web. Thus, theoretically, any existing information management application could be configured to be a Semantic Web application. Meanwhile, specific Semantic Web-based applications could be developed for handling information more semantically. For example, the SWIMS search engine allows end-users to select ontological concepts and properties as search criteria, so as to identify the exact information that the user is looking for. An interface template of a SWIMS document search engine is presented in Figure 7-40. The layout of the interface can be customised to meet the user's need. In the following sections some examples are presented for performing a series of SWIMS documented information management activities. These examples are based on Protégé, an ontology editing software, from which all the screenshots have been captured. It is not an end-user application, but can export information to any end-user's applications.

(2) From the Ontology Engineer's Perspective:

As analyzed in Section 6.3, in this study, Protégé is selected as the ontology editor because it provides an extensible ontology management environment and explicit methodologies for ontology development and management. Protégé allows

developers to configure ontologies and use various plug-ins, including ontology reasoners and mergers. Here, RacerPro and PROMPT are selected as the ontology reasoner and merger. Please refer to Sections 6.3 and 6.8 for more information about the software selection.

The functions of SWIMS are presented using IDEF0 diagrams. The IDEF0 notation is explained in Figure 7-1. An IDEF0 model indicates major functions/activities (called nodes) and the input, control, output, and mechanisms associated with each major function/activity.

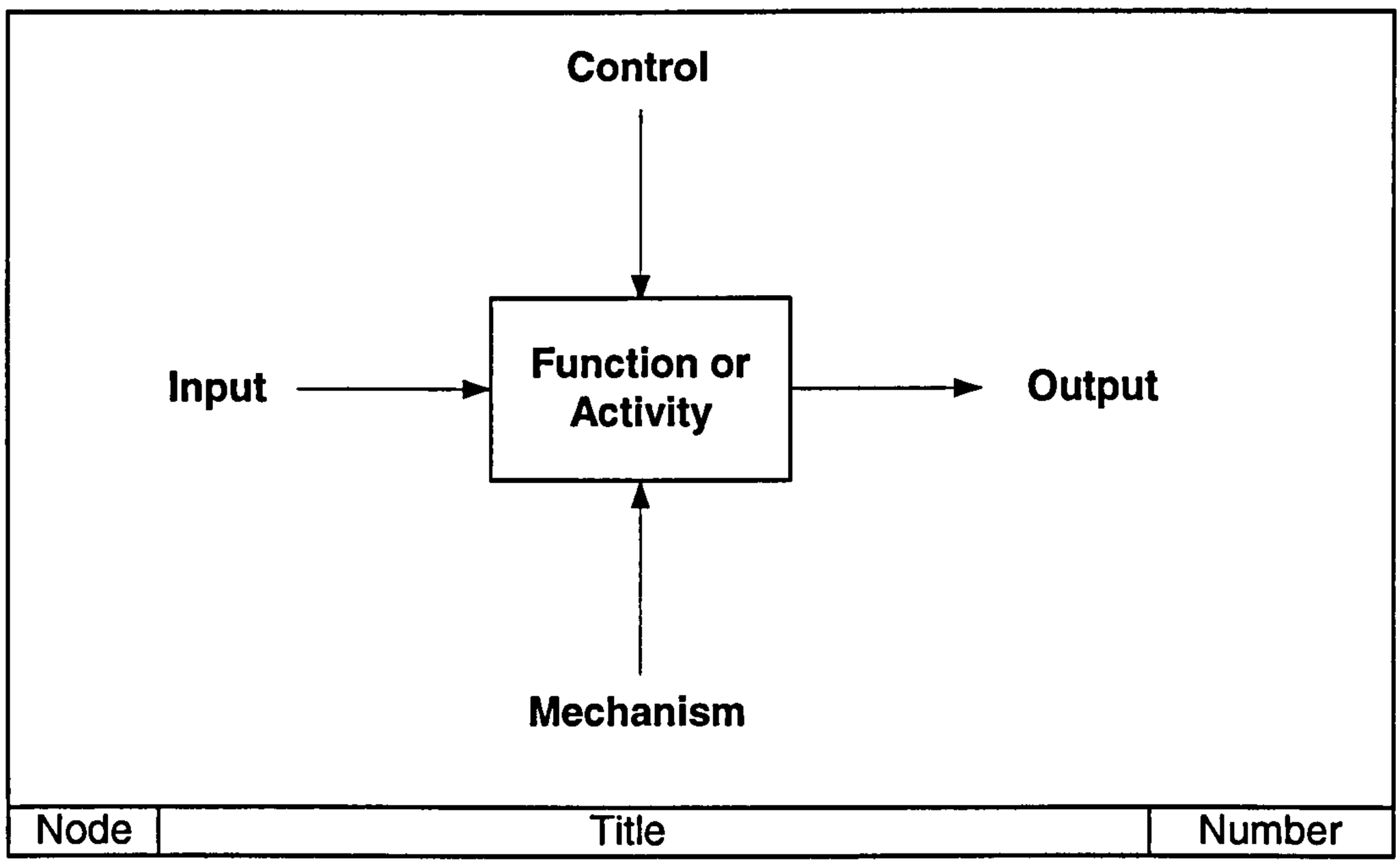


Figure 7-1: An IDEF0 Node ((Hanrahan, 1995))

IDEF0 nodes are combined into diagrams that describe activation of the functions/activities. An IDEF0 model supports the top-down decomposition of a function/activity. The hierarchy of diagrams is maintained via numbering the nodes. Figure 7-2 shows this with A-0, A0, A1, and A11 diagrams. Please refer to the *Integration Definition for Function Modeling* (NIST, 1993) for more information about the IDEF0 Process Modeling Methodology.

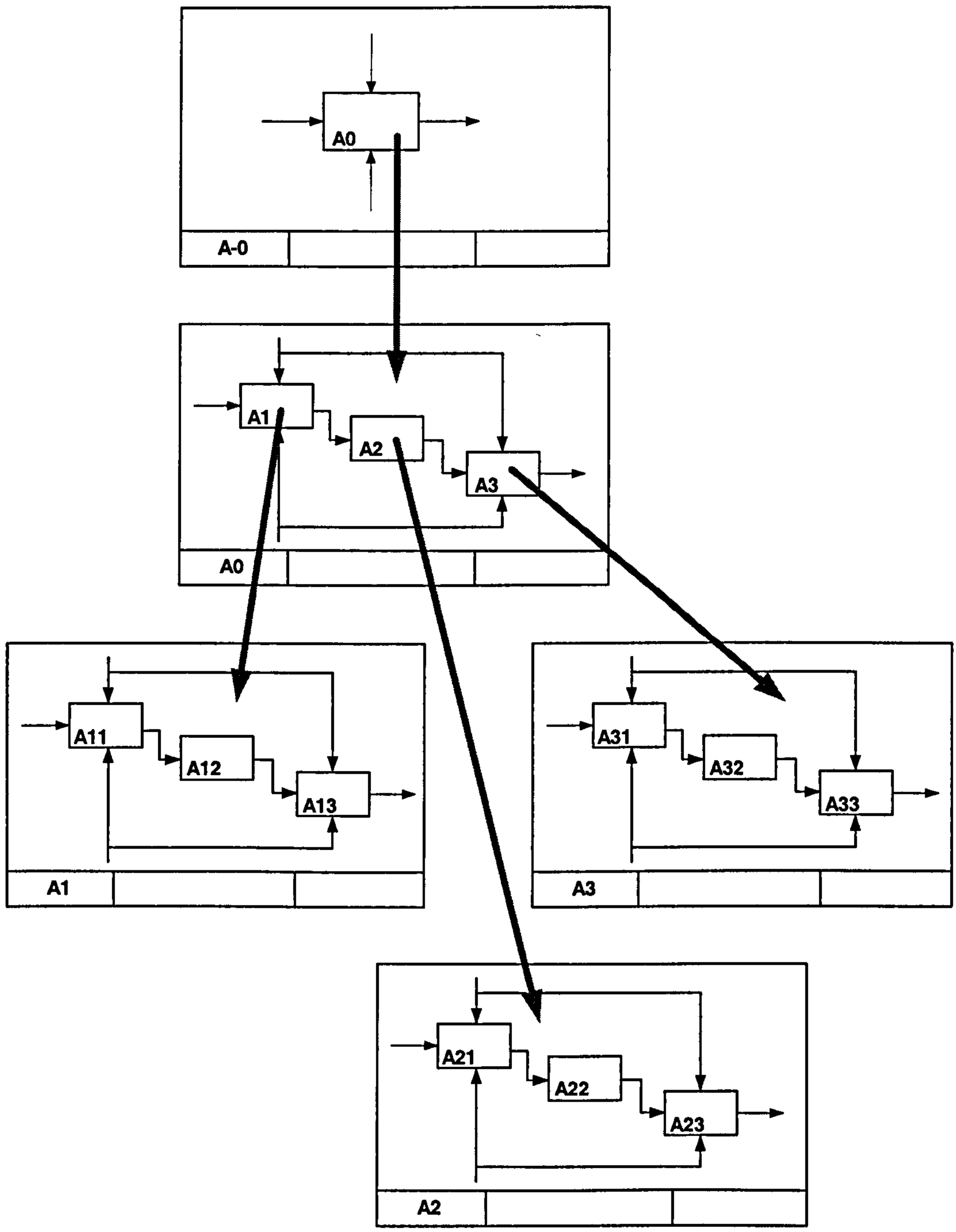


Figure 7-2: Hierarchical View of IDEF0 Model (Hanrahan, 1995)

7.3 PROTOTYPE AND DEMONSTRATOR

The scenario presented in Figure 5-3 is a typical scenario for managing construction project information in a Semantic Web environment. Project information is often wrapped in various documents. Traditional project information management approaches are at the document level, which cannot access the contents of

documents and leads to many problems, as argued in Sections 3.3 and 3.4. In this scenario, SWIMS is implemented as an information-centric document management application. Figures 8-3 and 8-4 illustrate the functions of SWIMS in this scenario. The task of processing documented project information is undertaken by IT tools, including hardware and software, as well as project partners. The management of project information is controlled by the dependencies of information, requirements from project partners, and logical rules. The construction project information repositories (consisting of product, process and actor's information), ontology libraries (both shared ontology and speciality ontologies), and the RFI (Requests For Information) from end-users are inputted into the SWIMS module. After a series of activities, the managed information (being stored in the information repositories or exported as project reports), updated ontologies (being stored in ontology libraries), alerting messages (being sent to related actors), and the responses to queries (being returned to the end-users) are outputted.

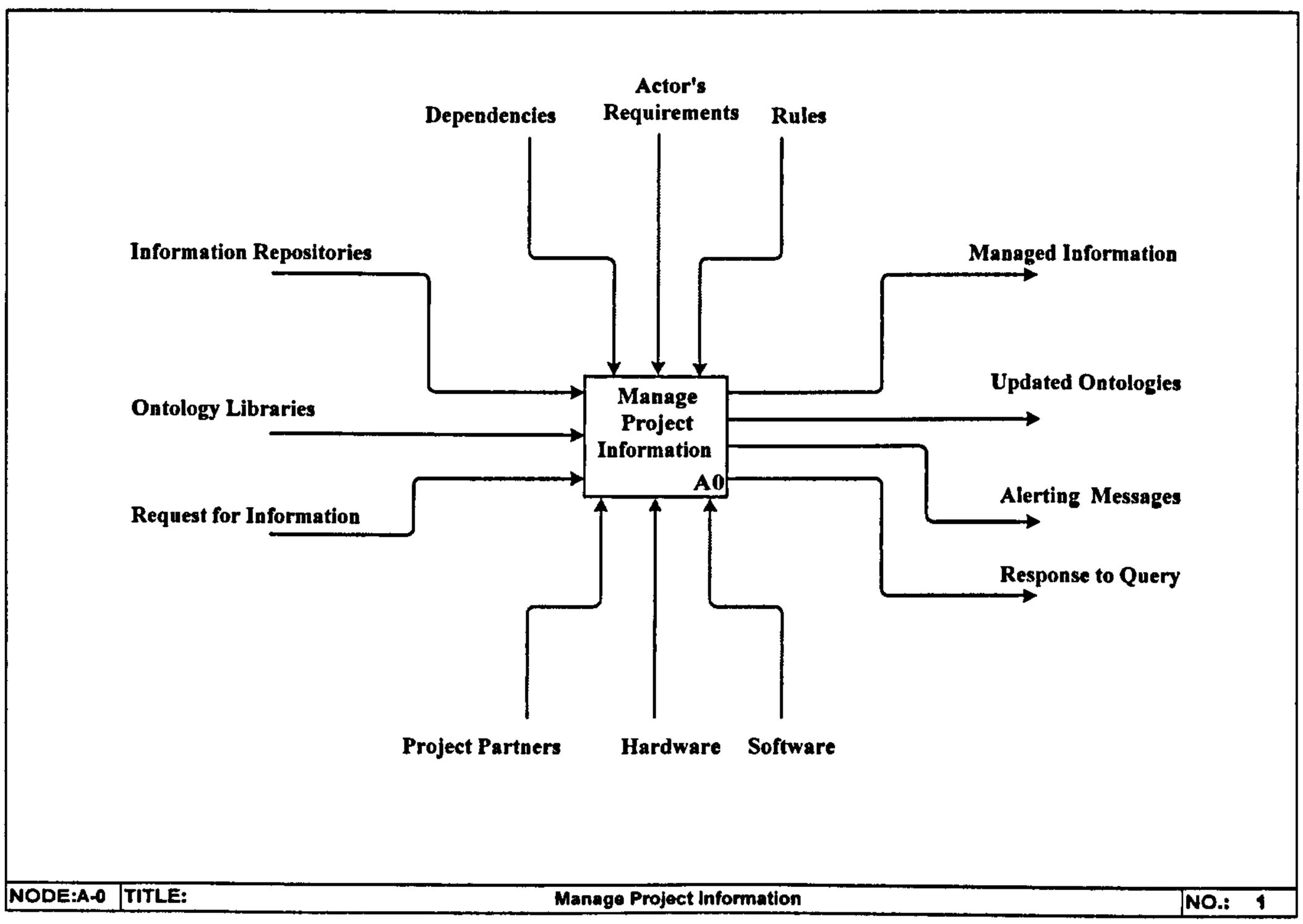


Figure 7-3: Construction Project Information Management Process in IDEF0-Node A-0

As represented in Figure 7-4, in SWIMS, the construction project information management activities is performed in three functional modules: the *manage*

ontology module, *manage actor information* module, and *manage documented information* module. Based on the ontology from the m*anage ontology* module, IT tools handle the information about actors or information contained in project documents, so as to respond to the RFI. In the *manage documented information* module, some suggestions for changing the ontology or user profile are generated, which will be fed back to the *manage ontology* and *manage actor information* modules.

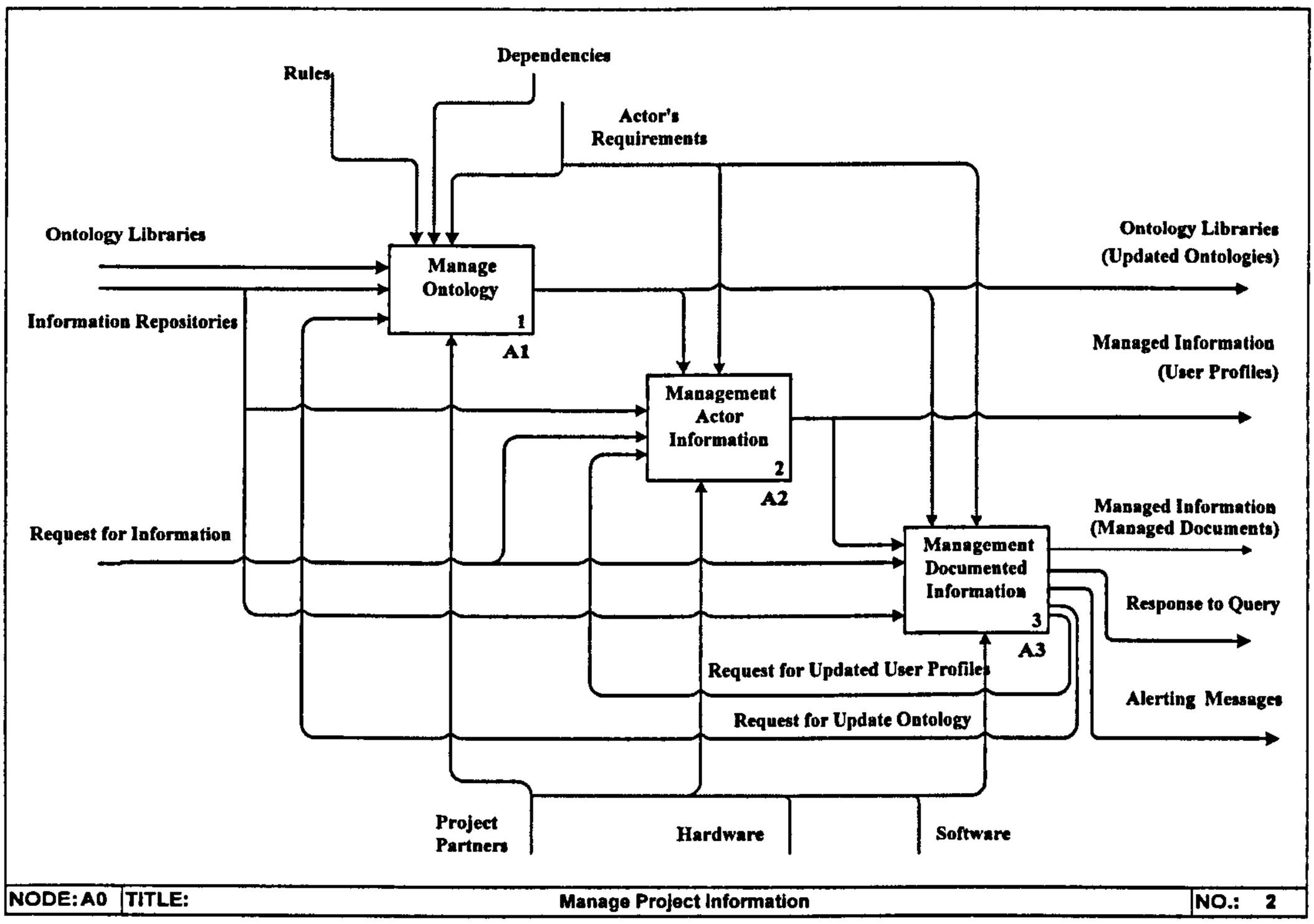


Figure 7-4: Construction Project Information Management Process in IDEF0-Node A0

7.3.1 Management of Ontology

The use of ontology and rules distinguishes SWIMS from traditional information management systems. The rules are defined by ontology elements (concepts, properties and relationships) whilst each ontology element is described in an RDF statement using XML syntax (refer to Section 2.2). They work together to enable computers to access the meaning of information and process it. In short, ontology is the key in representing the semantics of information.

The SWIMS Ontology consists of two parts: the shared ontology and speciality ontologies, which are developed in Chapter 6. Activities about the management of the SWIMS Ontology can be illustrated in Figure 7-5, which consists of four activities: *develop ontology*, *merge ontology*, *update ontology*, and *remove ontology*.

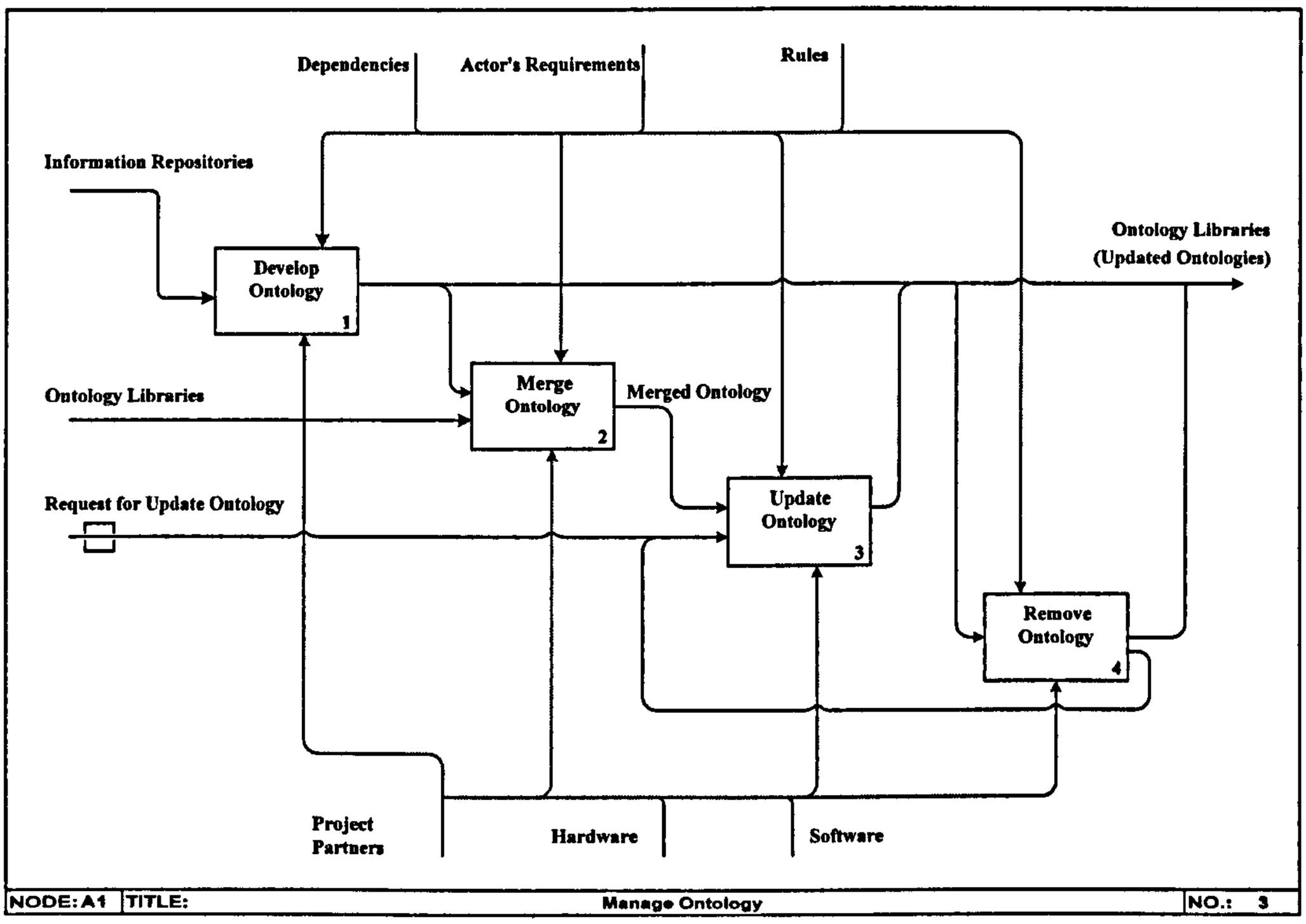


Figure 7-5: Ontology Management Process in IDEF0- Node A1

(1) Develop Ontology

The SWIMS Ontology has been developed in Sections 6.4 to 6.6. In this demonstration, there are three ontologies to be used: SWIMS Shared Ontology 1.0, SWIMS Product Ontology 1.0 and SWIMS Document Ontology 1.0. The first ontology is a shared ontology and last two are speciality ontologies. The class hierarchy of each ontology is listed in the appendixes.

SWIMS Shared Ontology 1.0 contains seven top-level concepts (actor, control process, product, project and technical topic), and their sub-concepts within three classes of hierarchy. It is compliant with the IFC kernel taxonomy. As captured in Figure 7-6, its concept class hierarchy is displayed on the left hand side of the window. On the right hand side, there are properties of the selected concept.

Document Related Technique is selected in Figure 7-6. This concept involves all issues *related to* a document. This definition is marked in the annotation property and defined by an object property in the middle right hand side of the window. On the lower right hand side of the window, the upper classes of Document Related Technique and its disjointed concepts are listed. If a concept is disjointed from the selected concept, they cannot share any common instance. This concept has not yet been specified as disjointed from any concept.

SWIMS Product Ontology 1.0 is an ontology about detailed physical and non-physical elements of product and other product-related concepts (such as structural element, equipment, material, and temporary element) and their properties and relationships. It is compliant with IFC core extension (ifcProductExtension) and, probably, several IFC domain/application (i.e. ifcArchitectureDomain, ifcStructuralAnalysisDocmain, etc.) taxonomies and terms in related UniClass facets (refer to Section 6.5-Step 4-(1)). The concept class hierarchy of SWIMS Product Ontology 1.0 is displayed on the left hand side of Figure 7-7. Concept structural element is selected and its properties are displayed.

SWIMS Document Ontology 1.0 contains concepts, properties and relationships related to the document concept, which are displayed in Figure 7-8. It is not necessary that a speciality ontology includes all top-level concepts. Only those related to the specific domain are involved in the speciality ontology. So SWIMS Document Onotology excludes some top-level concepts, such as control and technical topic. But it includes the sub-concepts of document related technical topic, such as the present medium and subject content of documents.

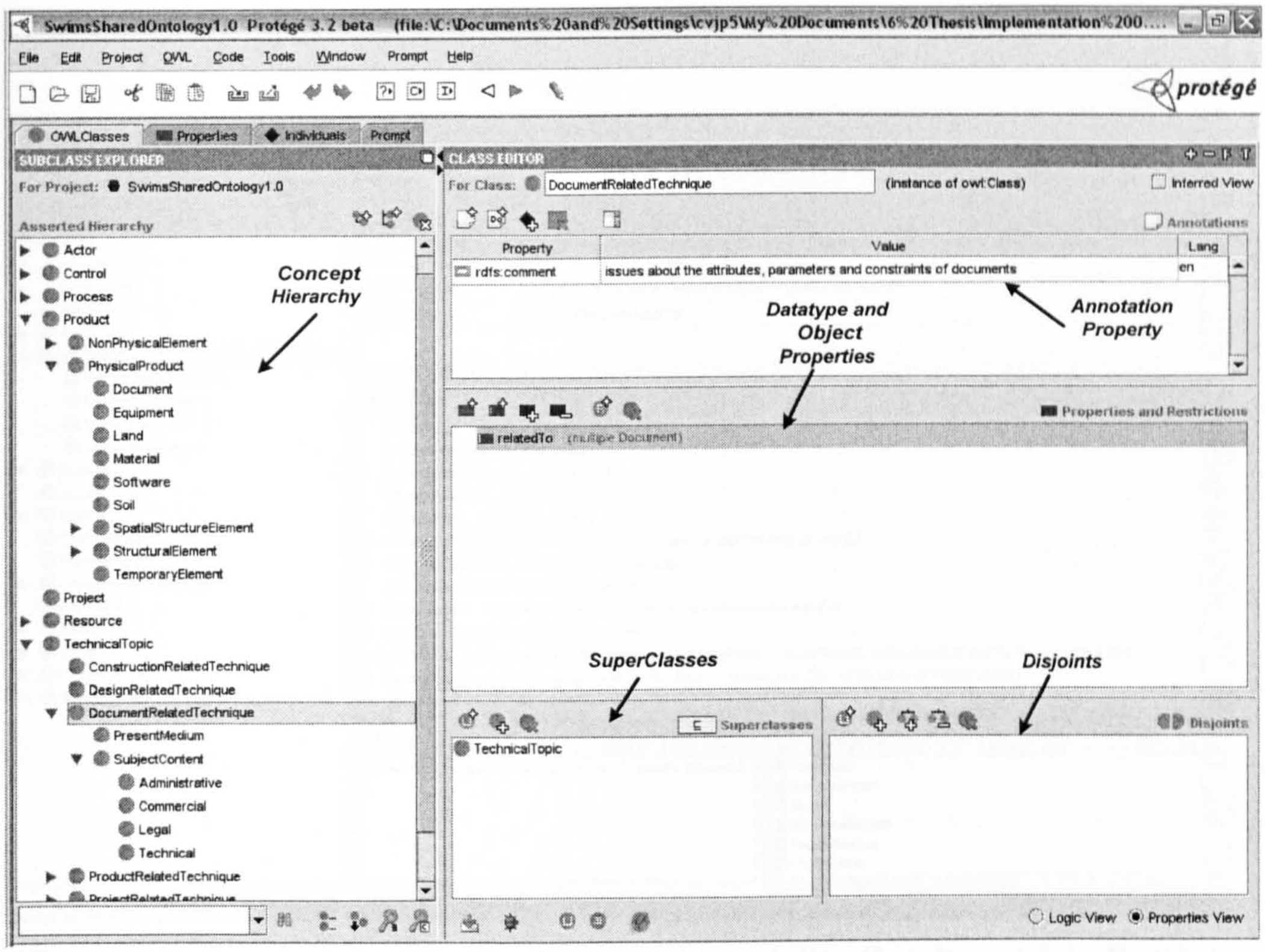


Figure 7-6 : Screenshot of SWIMS Shared Ontology 1.0 in Protégé

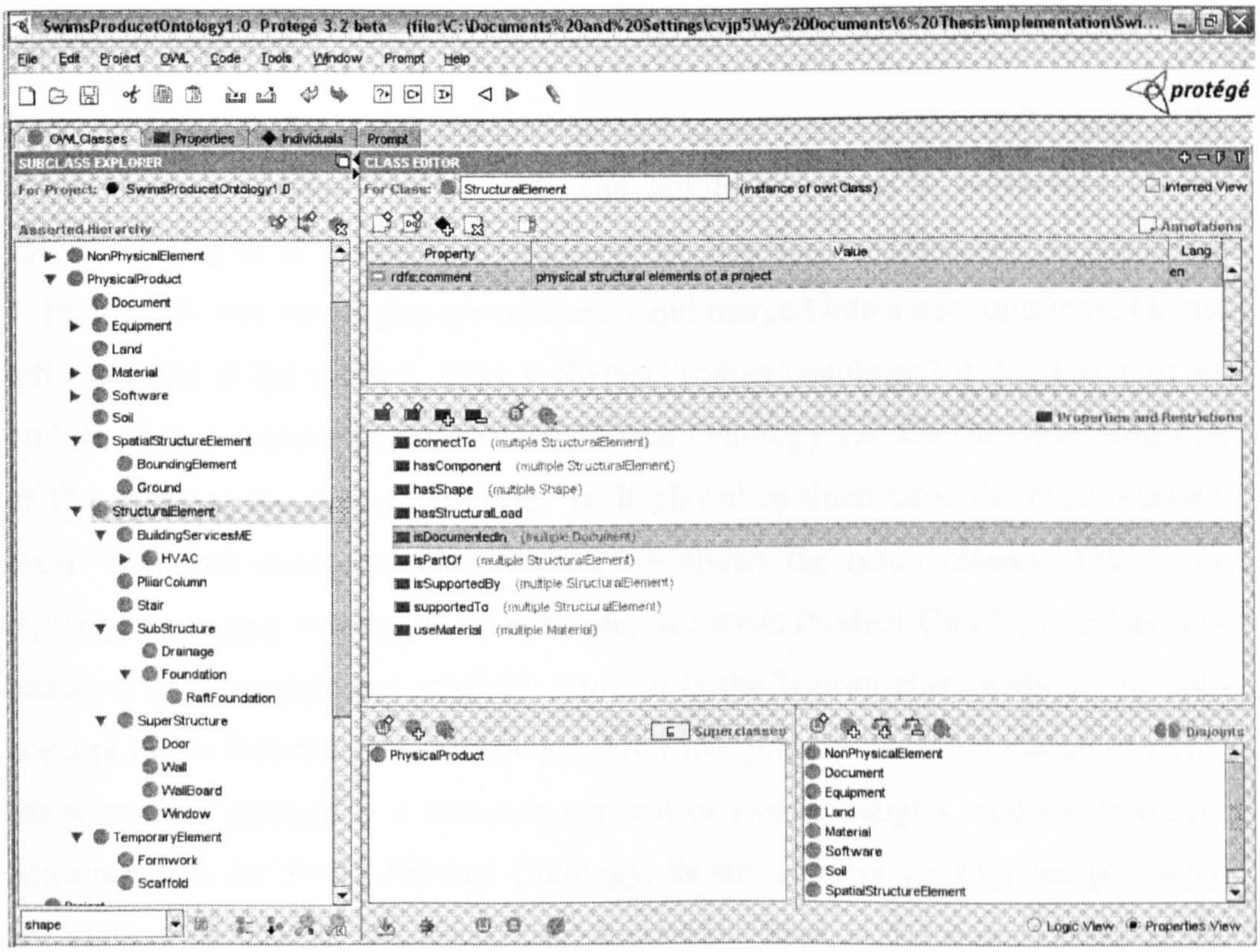


Figure 7-7: Screenshot of SWIMS Product Ontology 1.0 in Protégé

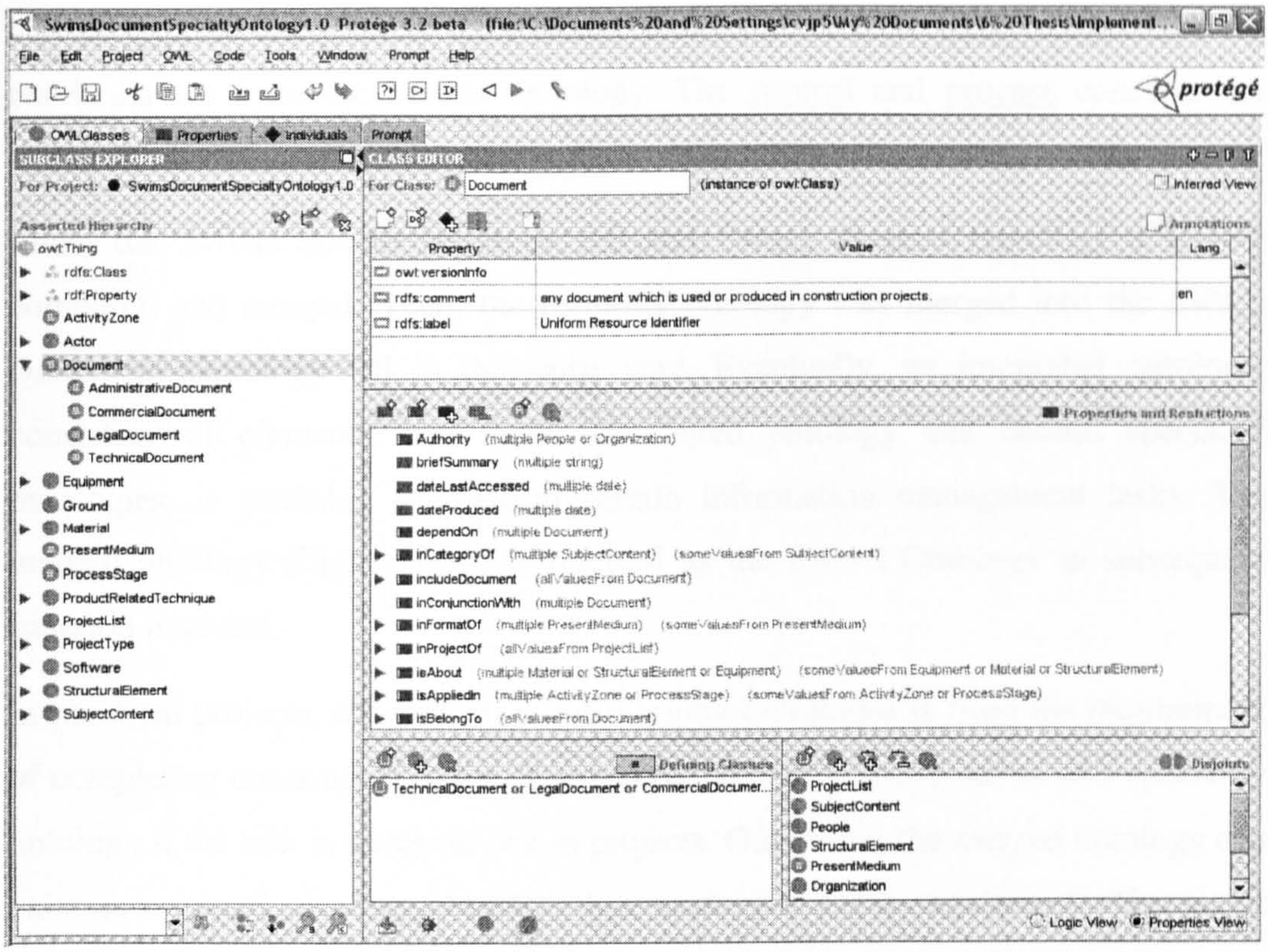


Figure 7-8: Screenshot of SWIMS Document Speciality Ontology 1.0 in Protégé

(2) Merge Ontology

In the application for managing documented construction project information, the above three ontologies are merged. The ontology merger is performed by the PROMPT plug-in in the Protégé ontology development environment. As displayed in Figure 7-9, two ontologies are compared and merged into a new ontology. On the left hand side of the window, there is SWIMS Product Ontology 1.0. Next to it, in the middle of the window, there is SWIMS Shared Ontology 1.0. On the right hand side of the window, the merged ontology is displayed in three tabs: the result classes, result slots and result instances. Figure 7-9 shows the result classes. Taking the temporary element concept as an example, in SWIMS Product Ontology, it has sub-concepts of formwork and scaffold, while it is the bottom class without any sub-concept in the SWIMS Shared Ontology. After merging, in the result class hierarchy, the temporary element is a common concept of two ontologies, and has two sub-concepts from the SWIMS Product Ontology. Its sub-concept scaffold and formwork are marked in blue in the result class hierarchy. Blue is the same colour as the Product Ontology on the left hand side of the window. Also, the concepts control

and process do not exist in the Product Ontology but have been merged into the result classes from the Shared Ontology. The control and process concepts are marked in red, the same colour as the Shared Ontology.

Thus, the SWIMS Shared Ontology 1.0 and SWIMS Product Ontology 1.0 were compared and merged. Then, the resulting ontology was merged into the SWIMS Document Ontology 1.0 in the same way. Eventually, an integrated ontology, consisting of elements from both the shared ontology and certain speciality ontologies, is available to perform certain information management tasks. The merged ontology (Figure 7-10) is referred as the SWIMS Ontology in subsequent function modules.

In practical projects, the motivation of merging ontologies is from the requirement of completing construction tasks. The merged ontology can be saved as a speciality ontology if the task is a regular one in projects. Otherwise, the merged ontology can exist as a virtual ontology, which is just used once for a certain task. The major objective of merging ontologies is for collaboratively using existing ontologies and not for generating a new ontology. Therefore, not all merged ontologies are saved in the ontology libraries.

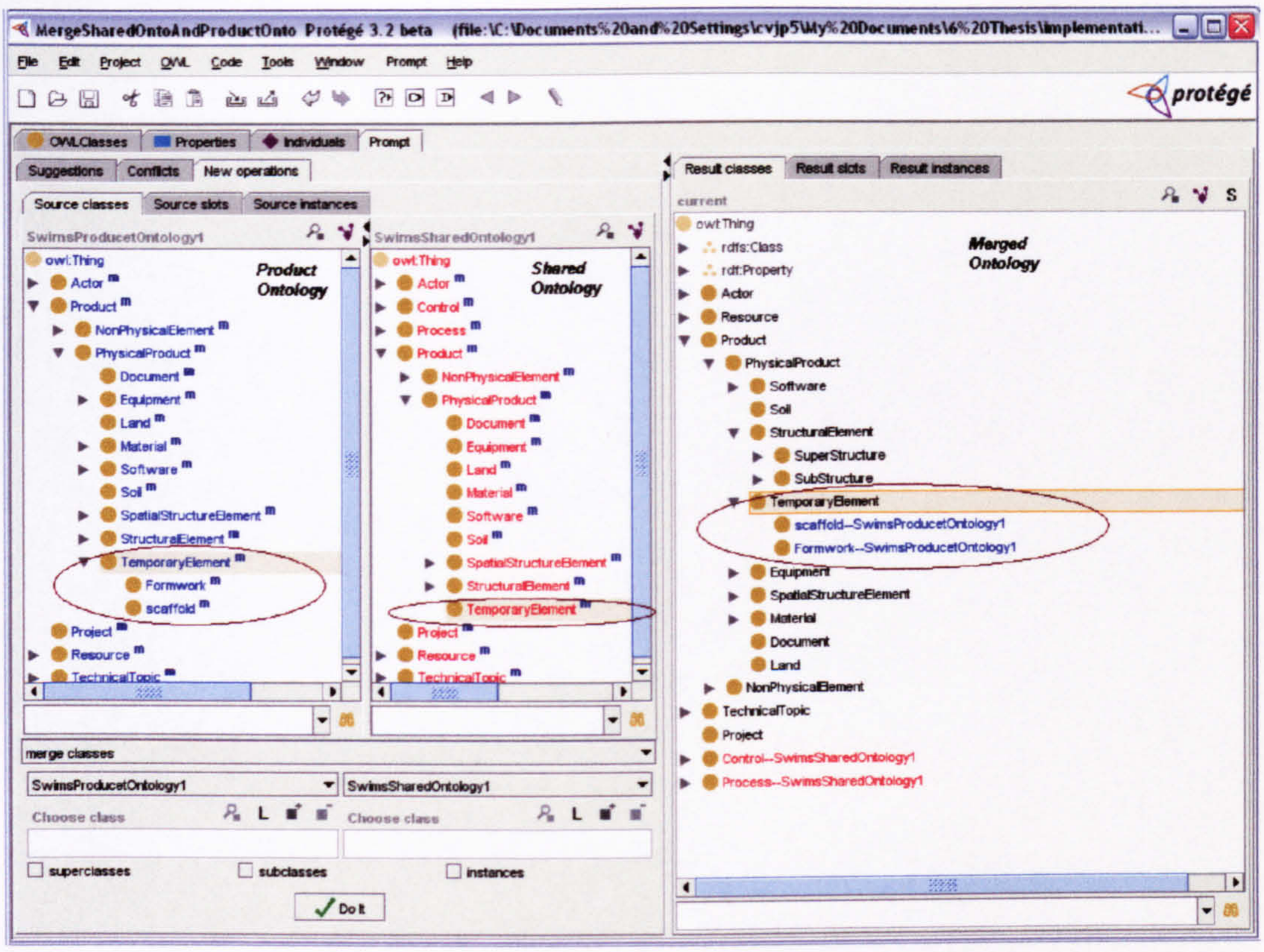


Figure 7-9: Screenshot of the Merging Shared Ontology with Product Ontology

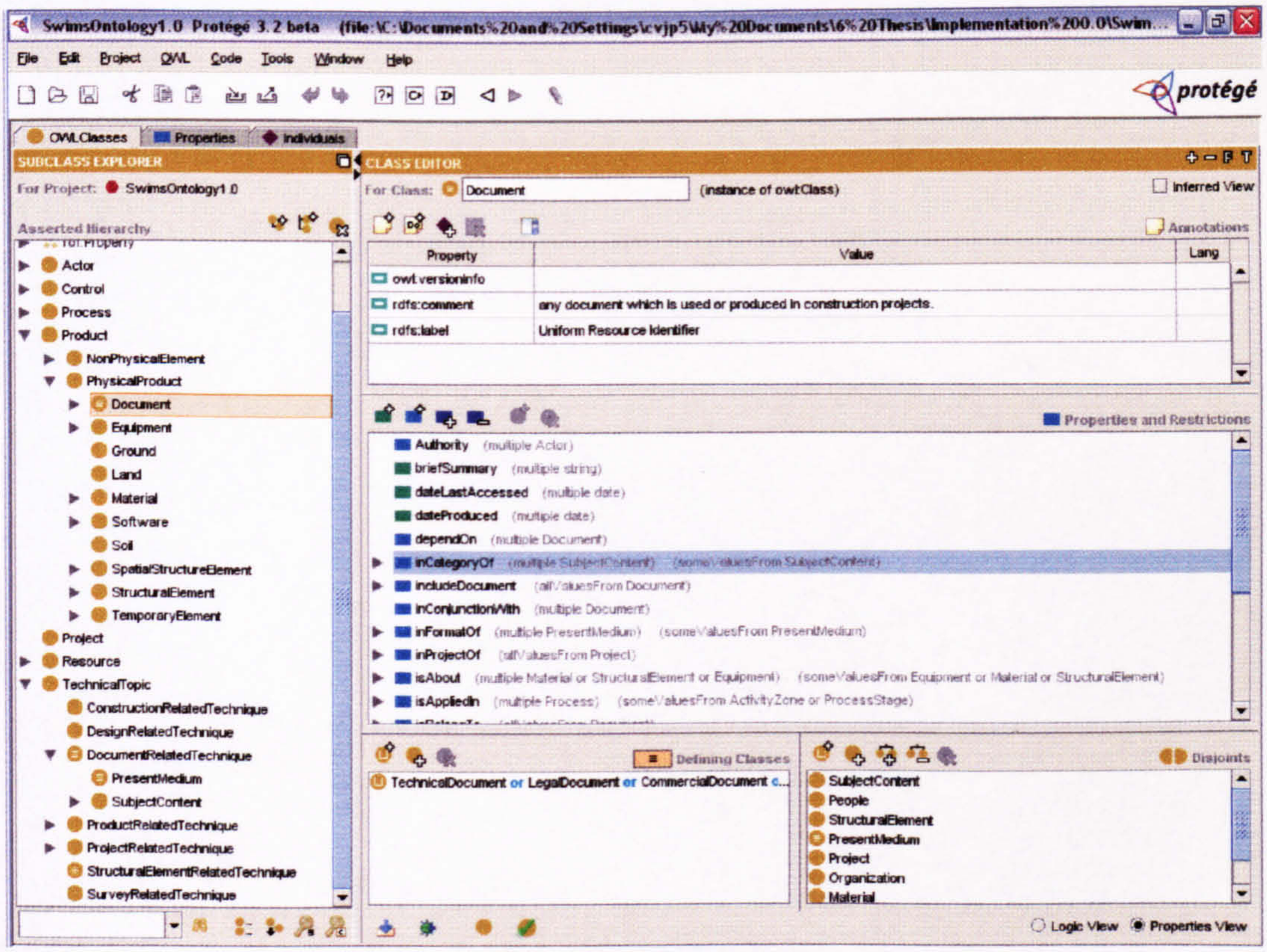


Figure 7-10: Merged SWIMS Ontology

(3) Update Ontology

The merger of ontologies aggregates and restructures ontology elements but does not make any change to the elements. However, in the use of an ontology, new concepts, properties and relationships are identified and required for changing the original ontology. Therefore, the ontology has to be updated to meet these changes. The most important issue in the ontology evolution is keeping the new elements consistent with the existing ones. OWL reasoners are available for checking consistency in an ontology and suggesting changes to the taxonomy. Figure 7-11 is an example of adding new concepts into an ontology for the Royal Festival Hall Extension Building Project. RacerPro is used to check the consistency of the ontology and classify the taxonomy in the Protégé ontology development environment. After computing, some suggestions for change are listed in the bottom of the window. Accepting reasonable suggestions, the ontology will be updated and used in future work.

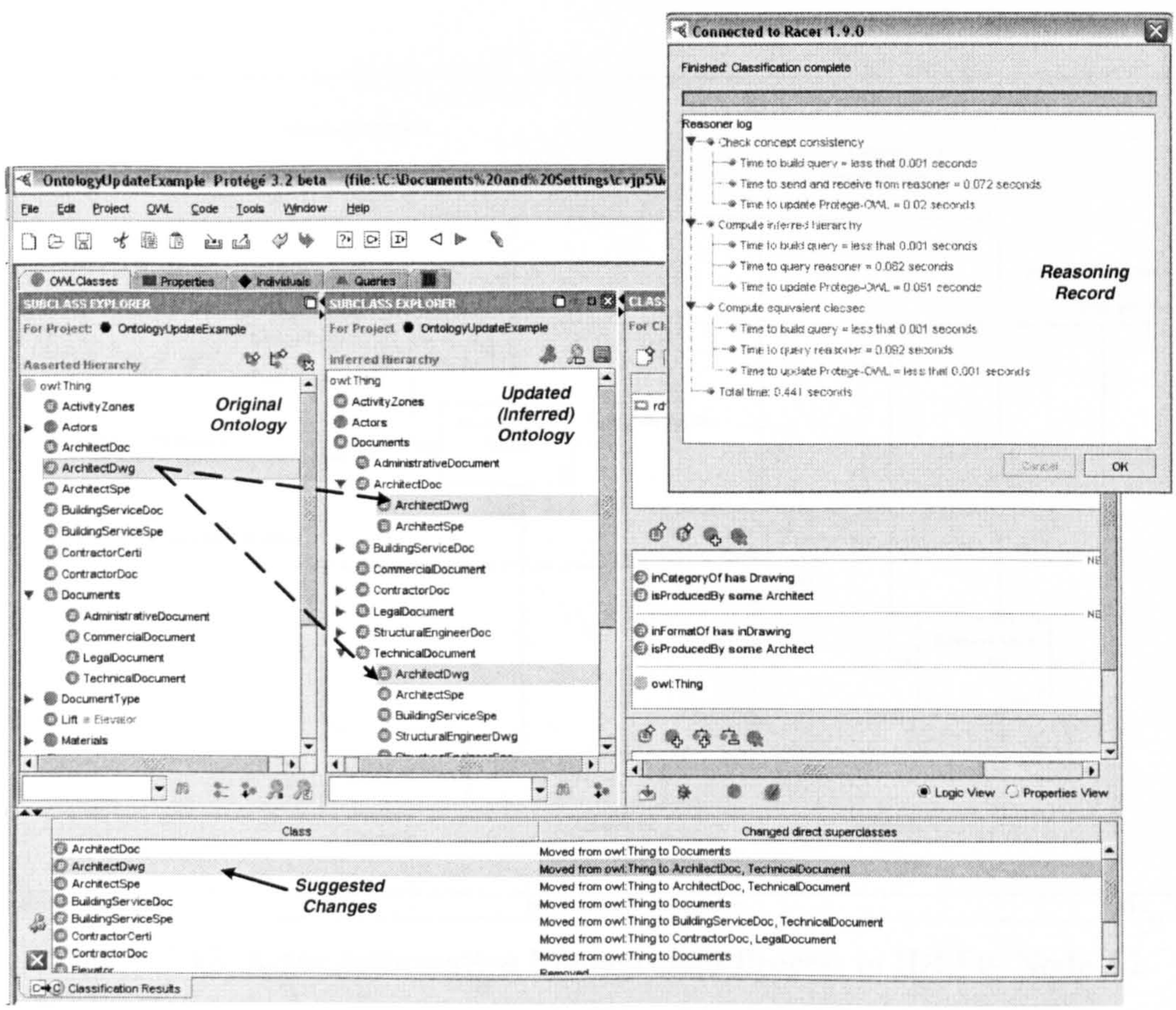


Figure 7-11: Check Consistency and Update Ontology in Protégé

(4) Remove Ontology

Outdated ontologies can be removed from the ontology library. It is as simple as deleting a file from a hard disk. Unnecessary elements of an ontology can also be extracted and removed. But it is a complicated process because the elements in an ontology are intertwined. After removing some elements from an ontology, it is necessary to check the consistency of the ontology and to update it. Sometimes, it requires professionals to manage the changing of an ontology.

7.3.2 Management of Actor Information

The concept actor in SWIMS represents all partners involved in the construction project. It could be an organisation or a human being. Information about people and organisations are somewhat different but the information management processes are the same. As depicted in Figure 7-12, information about actors is managed in four activities: *create user profile*, *manage user profile*, *update user profile*, and *remove user profile*.

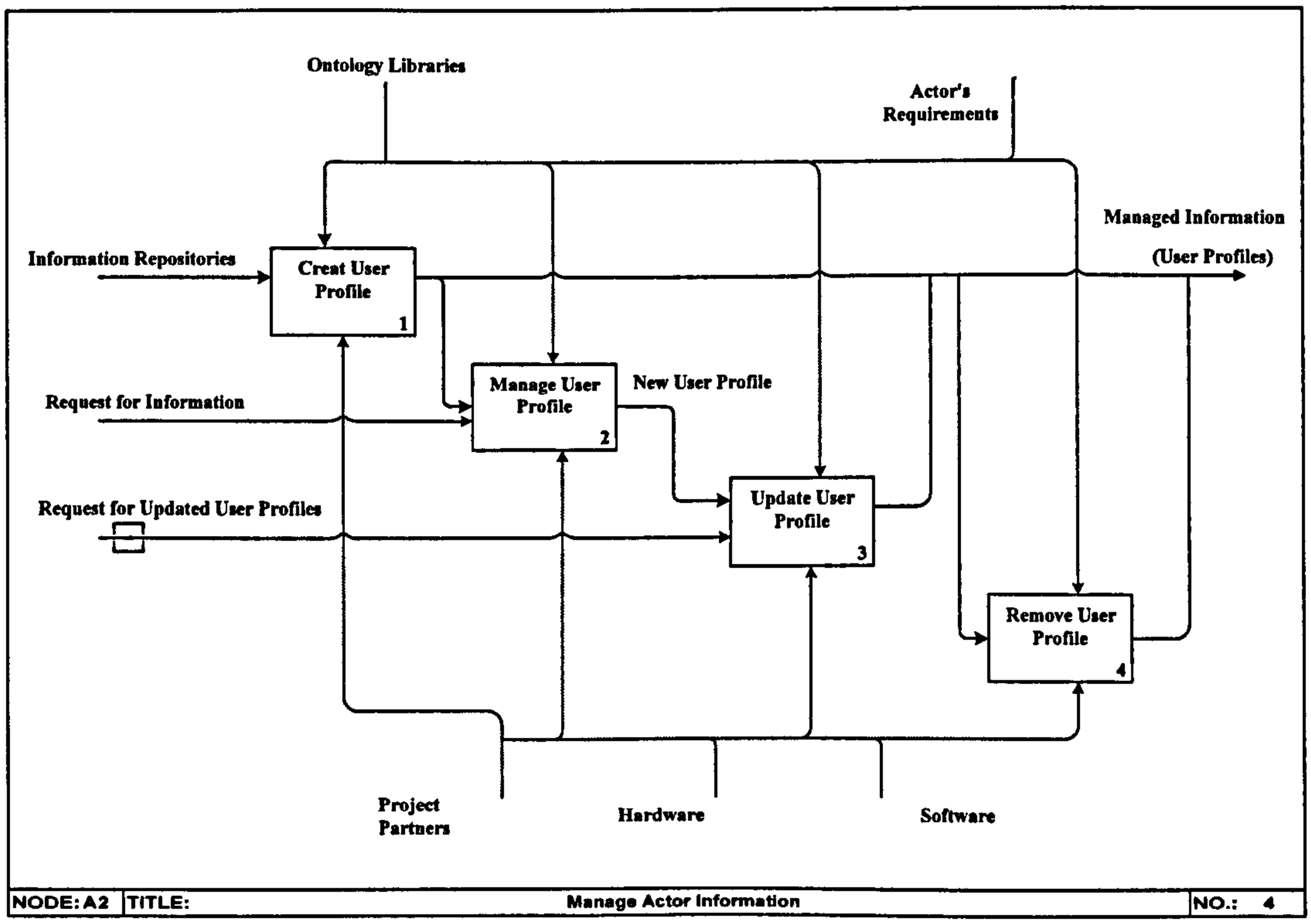


Figure 7-12: Actor Information Management Process in IDEF0- Node A2

(1) Creation of User Profile

Actors are annotated by user profiles in SWIMS. Taking the Royal Festival Hall Extension Building project as an example, its partners involve: customer, principal contractor, architect for RFH (customer) part, structural engineer for RFH part, architect and structural engineer for TWC (principal contractor) part, services consultant for RFH part, services consultant for TWC part, quantity surveyor and cost consultant, and building controller. The information on these partners, such as their contact person, telephone number, address, fax, e-mail, and related documents, is extracted from their own documents into user profiles, as proposed in Section 5.4.3. Figure 7-13 presents the annotation of partners using Protégé. On the left hand side of the window, the concept class hierarchy is displayed. Selecting the people concept, its annotation items (a property set) are displayed on the right hand side of the window.

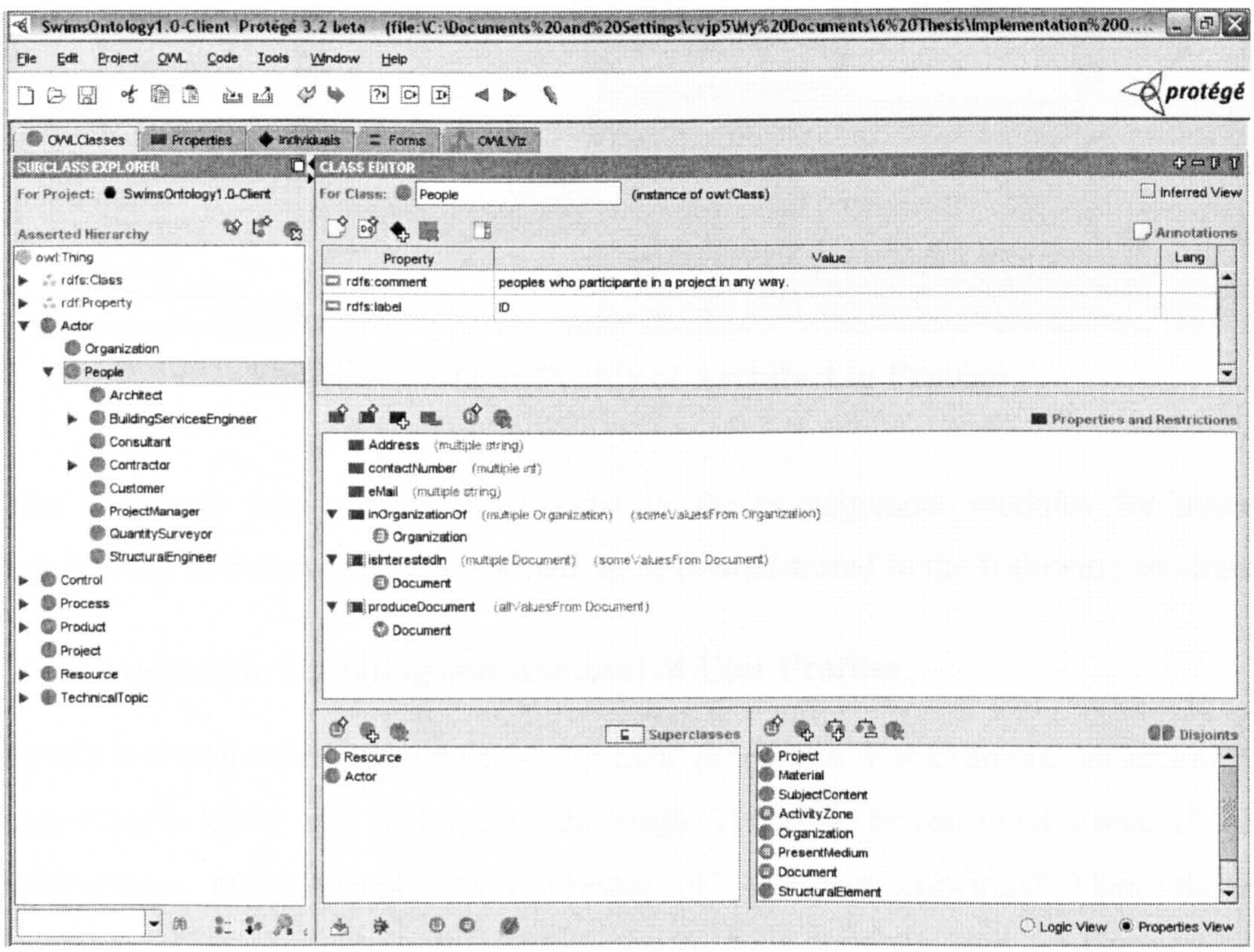


Figure 7-13: Screenshot of People Class and Properties in Protégé

Each project team member is an instance of the actor concept or sub-concepts. Information about them is extracted and filled into the value slots of each concept's

properties, as captured in Figure 7-14. For example, in the Royal Festival Hall Extension Building project, instances of the concept architect (highlighted on the left hand side of the window) are listed in the middle of the window. The highlighted instance's information has been extracted into the property slots on the right hand side of the window (marked by circle).

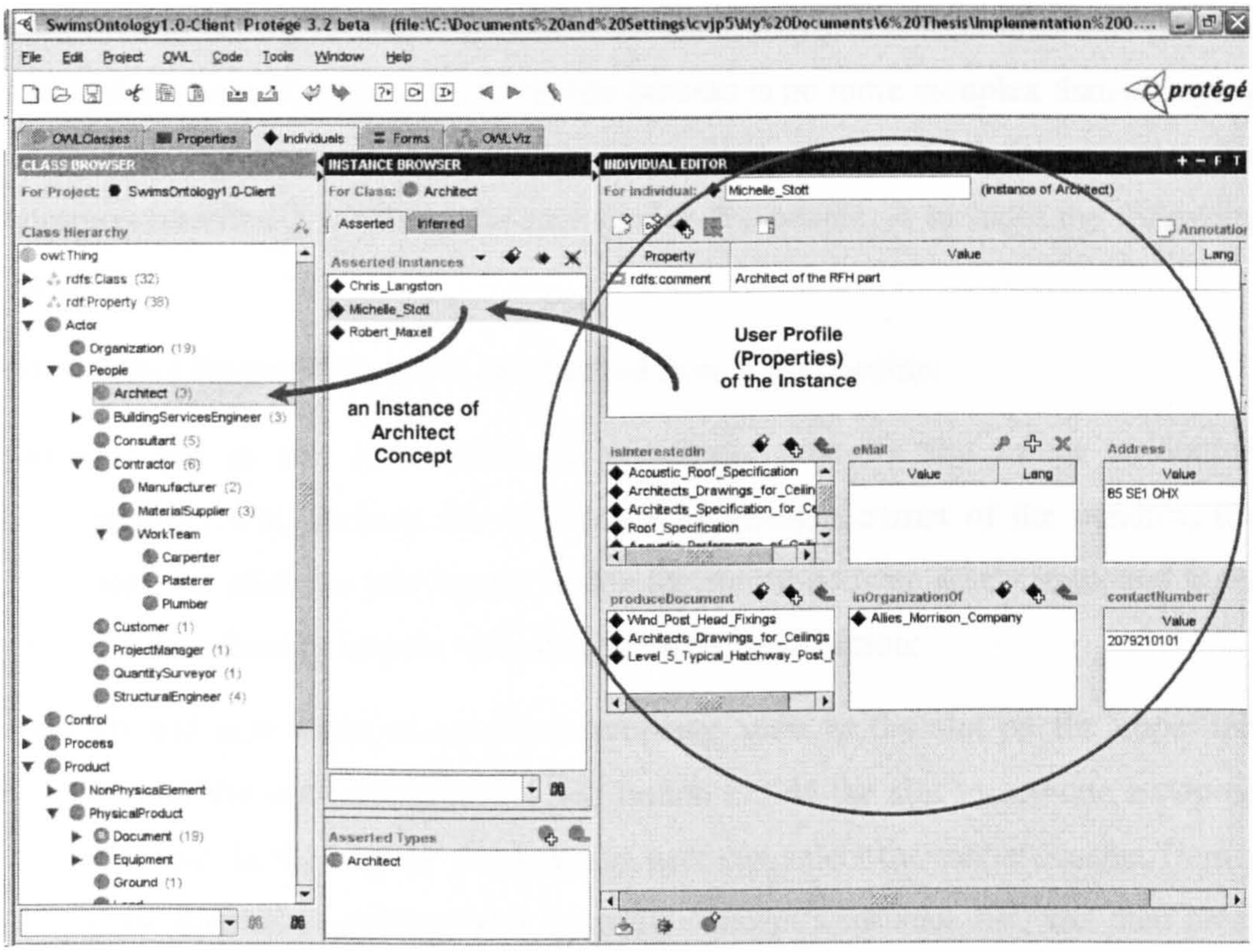


Figure 7-14: User Profile of Architect in Protégé

The annotated user profiles were used in the management modules for actor information and documented information, as demonstrated in the following sections.

(2) Management, Updating and Removal of User Profiles

SWIMS end-users can retrieve actors by their properties. For example, an architect may want to know who will use his drawings. Therefore he can input a request for information: which people "*is interested in*" architect drawings? Then, those instances of people with architect drawings in their "*is interested in*" slot will be retrieved. In other words, actors with the same property values constitute a virtual group. This leads to many additional benefits. For example, actors can easily access people, who have the same interests with him/her, then discuss and share

information with them. Also, when the common information is updated, computers can identify related actors and send alerts to them.

During the use of user profiles, project partners may find that the preliminary settings are not capable of satisfying their requests, such as a concept is related to their work but has not been included in their user profiles or they are no longer interested in certain concepts. In such a situation, they can correct their user profiles and update them. The user profile update process is no more complex than changing one's home address on an online bank account. Figure 7-15 demonstrates the common interface for editing the user profile for people; it includes the following steps:

(i) Select the property slot to be changed from a user profile;

(ii) To add or edit information in a datatype property slot or an annotation property slot, such as the one on the upper-right corner of the window, the user can click the *add* button beside the slot to activate a new item, and input the data directly into the slot; then click the *save* button;

(iii) To add new items to an object property, such as the slot on the upper-left corner, the user can click the *add* button beside the slot to activate a pop-up window. In the pop-up window, the user can select the target concept from a class hierarchy and instances from the concept's instance list, and then press *OK* and *save* buttons;

(iv) To remove an item from a property, the user can select the item to be removed, and click the *delete* button.

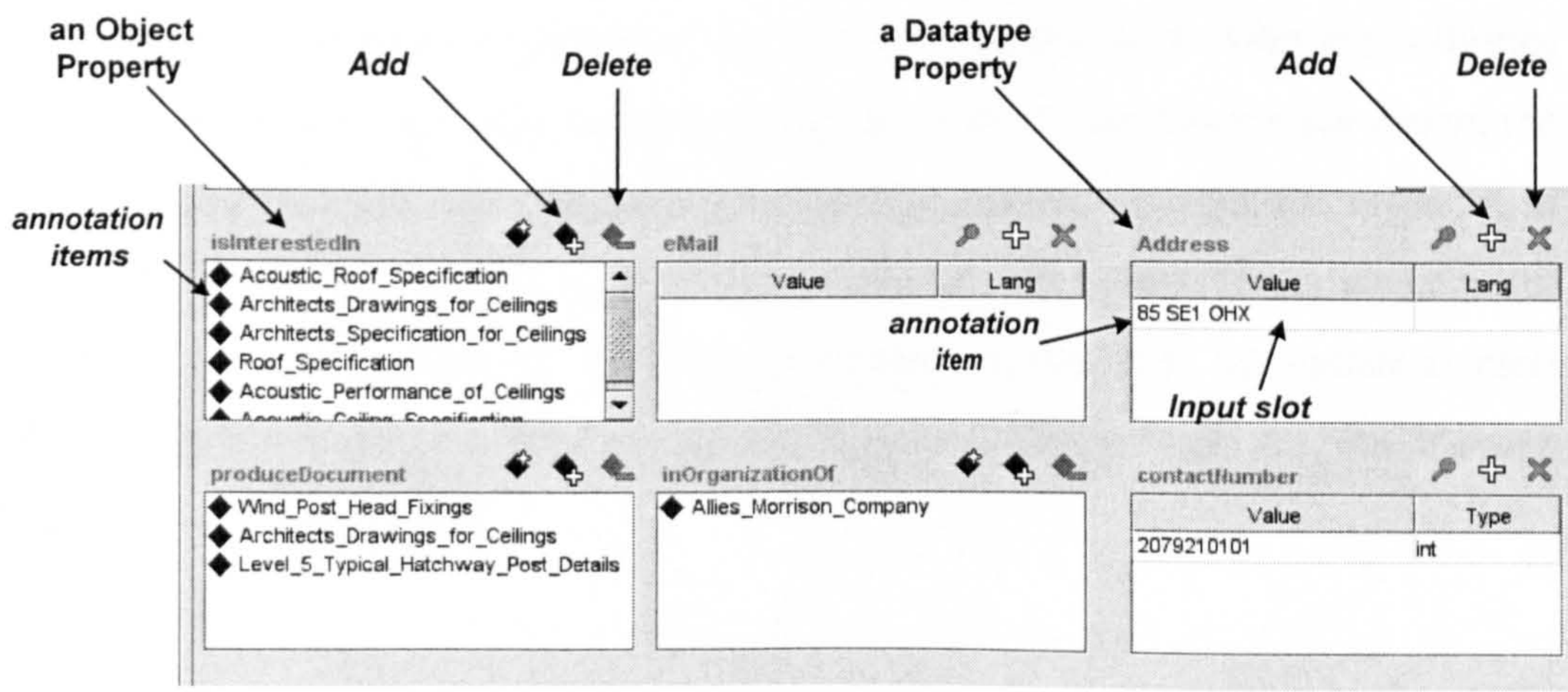


Figure 7-15: Edit Slots of User Profile

7.3.3 Management of Documented Information

A major task of this SWIMS application is to enable computers to access and process information documented in project files or databases. This task involves four activities as depicted in Figure 7-16.

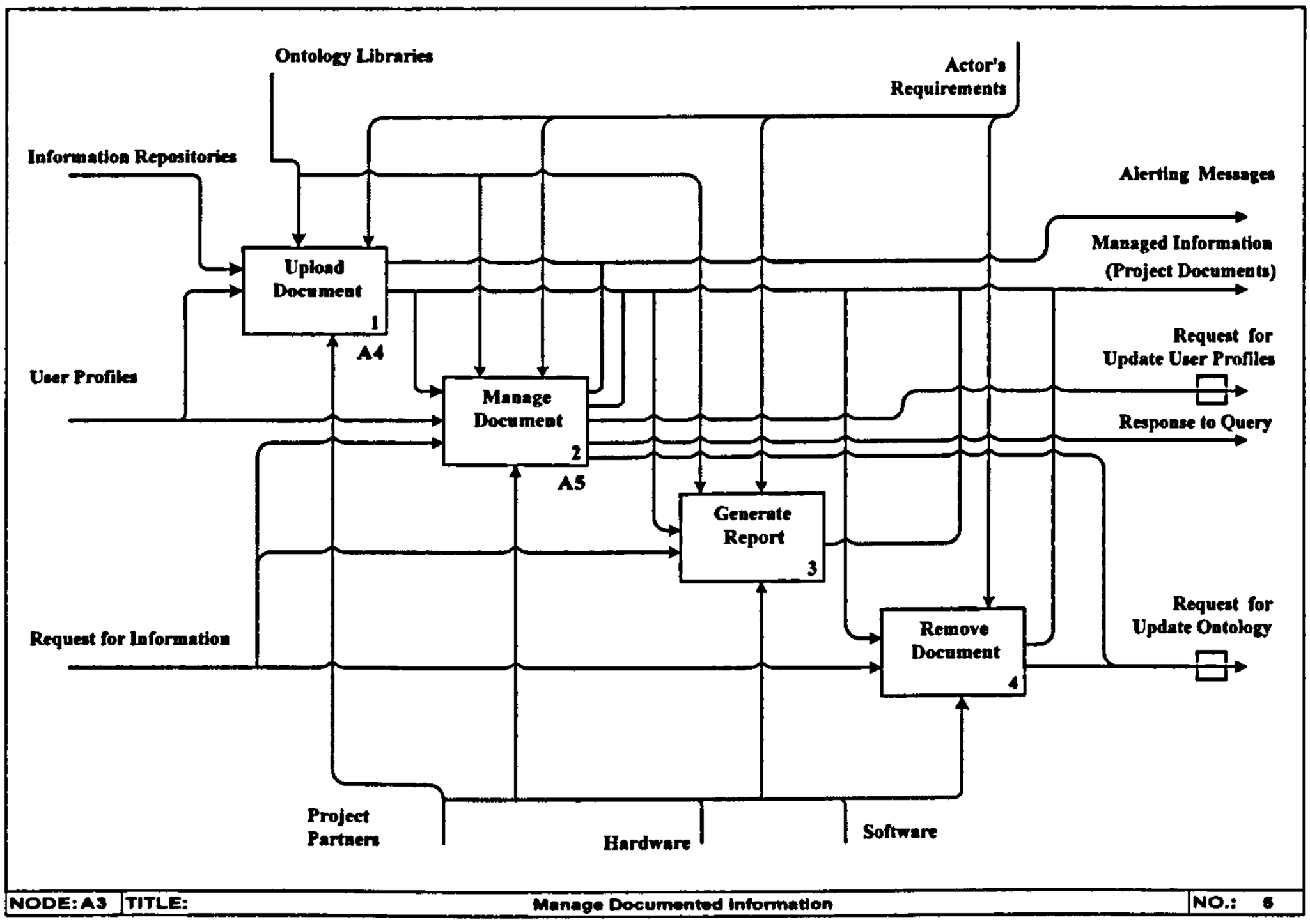


Figure 7-16: Documented Information Management Process in IDEF0- Node A3

The existing ontologies and actor's requirements are the constraints of this task. Project partners use IT tools to handle documented project information from information repositories, according to the user profiles and RFI. After the activities of *upload document*, *manage document*, *generate report*, and *remove document*, the documented information is managed. Also, project partners can get the response to their queries and update alerting messages of related documents. If any requests for the change of user profiles or ontologies are raised in this task, the update request will feedback to Node A1, the *manage ontology* module, or Node A2, the *Manage Actor Information* module.

(1) Document Upload

The document upload module involves three activities: *upload document*, *annotate document*, and *alert update*, which are depicted in Figure 7-17.

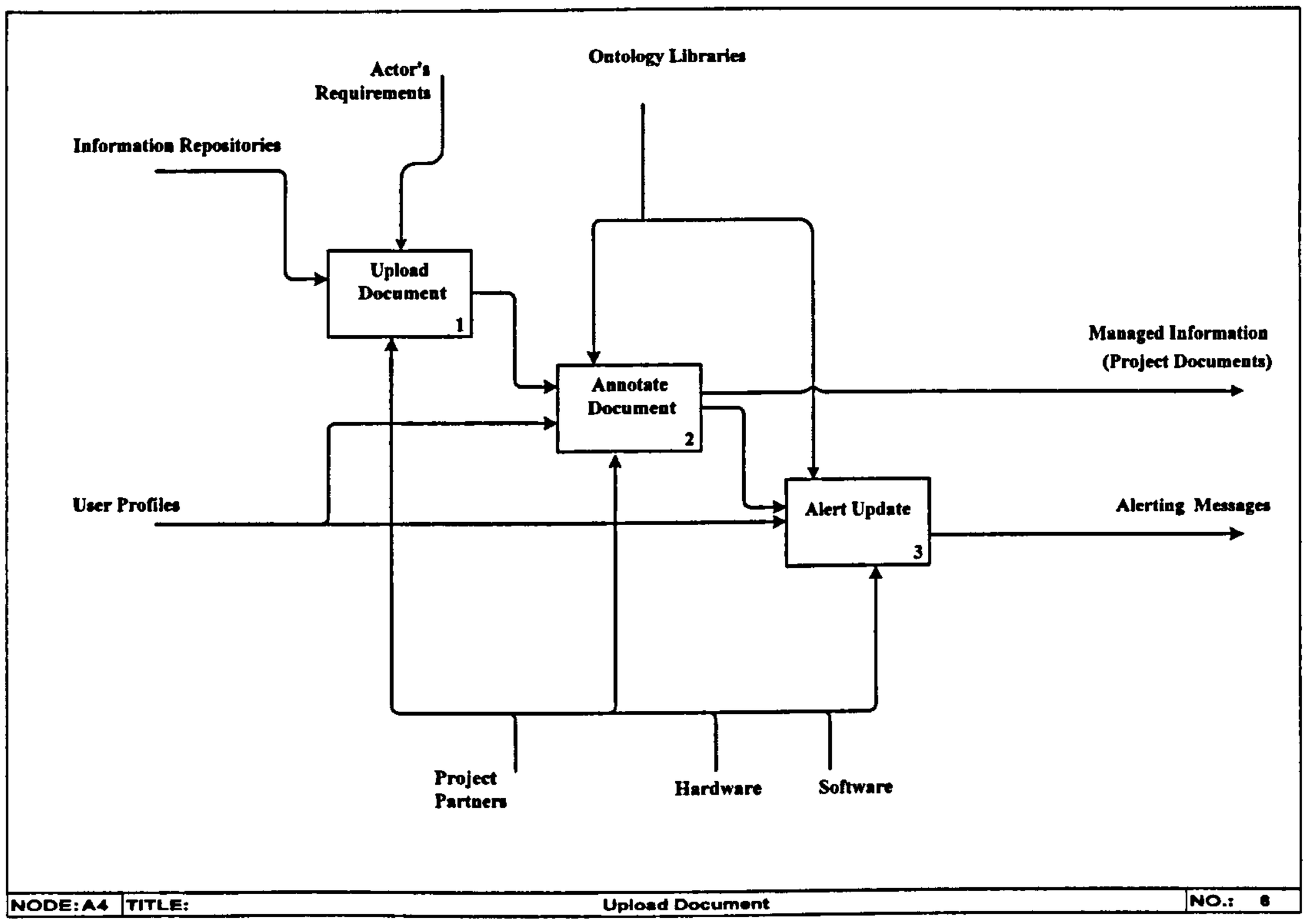


Figure 7-17: Document Upload Process in IDEF0- Node A4

In SWIMS, project information is uploaded from information repositories to SWIMS via documents or database files (being regarded as a type of document in this research). Every document is an instance of the concept document and linked to other instances, such as the instances of structural element or actor. Accompanied with the upload action, information contained in the document is extracted and annotated into the document's metafile.

A metafile model has been proposed in Section 5.4.2. A metafile includes the major properties of a document used in a certain application (the management of documented project information in this demonstration). Figure 7-18 is a screenshot of editing the metafile for a project document in Protégé. The value range of each property slot has been defined in the ontology development process. Please refer to Sections 5.4.2 and 6.5 for the definition of each property. Then, for each real project document, the annotation action includes the following steps:

(i) Select the property to be edited;

(ii) To edit the information in the slot of a datatype property, the user can click the *add* button beside the slot to activate a new item and input the data directly into the slot, then click the *save* button;

(iii) To add a new item to an object's properties, the user can click the *add* button beside the slot to activate a pop-up window. In the pop-up window, the user can select the target concept from a class hierarchy and instances from a concept's instance list, and then press the *OK* and *save* buttons;

(iv) To remove an item from the property, the user can select the item to be removed, and click the *delete* button.

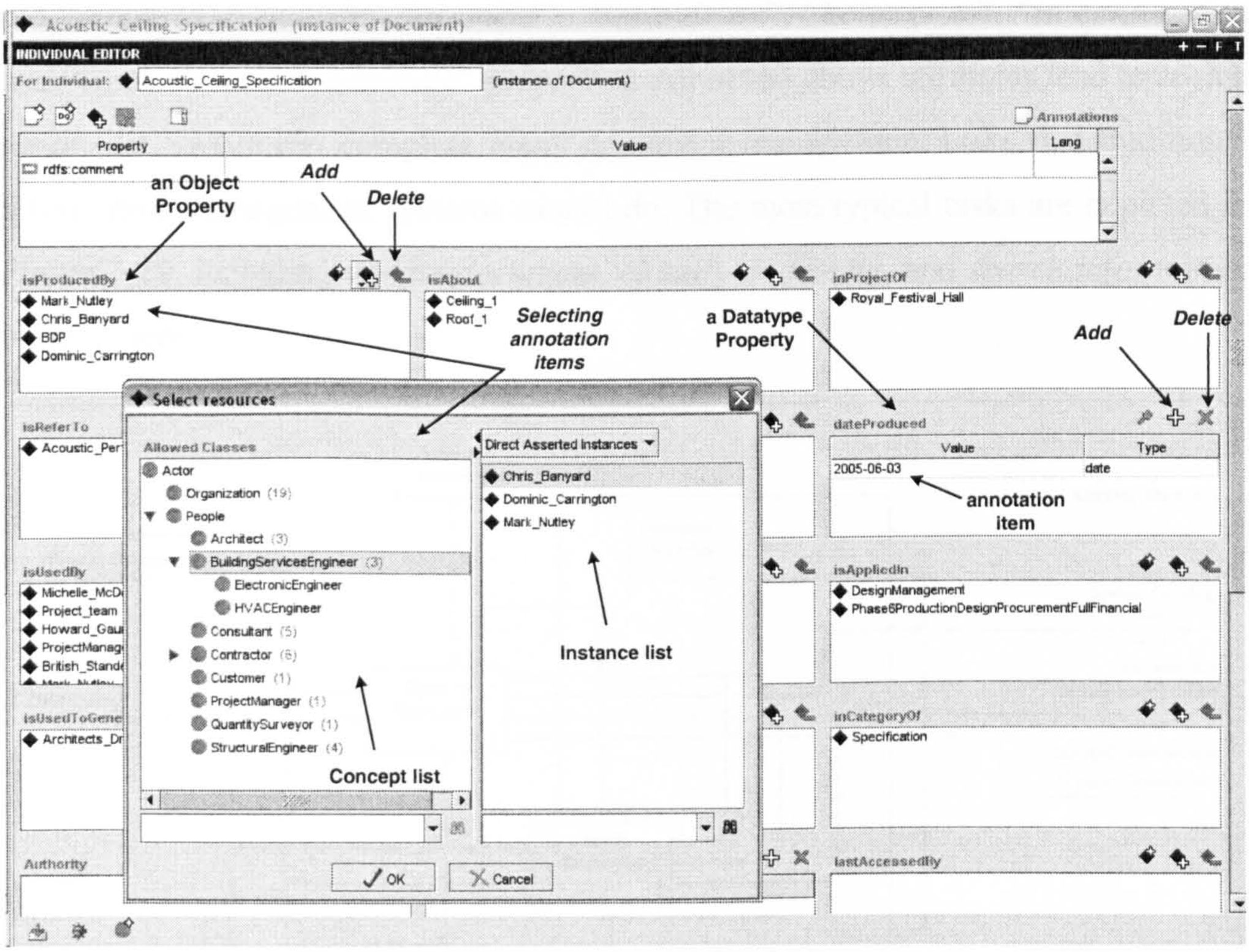


Figure 7-18: Metafile of a Project Document

In SWIMS, every document is linked to project partners by relationships such as "*is produced by*", "*is used by*", "*authority*" and "*last accessed by*". These relationships can be used in various circumstances. For example, once a document is uploaded on SWIMS or information in a document has changed, the users of the document will be informed by an update alerting message sent from the *alert update* function module.

Meanwhile, a project partner will also be notified if his individual properties, such as "is interested in" or "job" responsibility, match the "*is about*" property of the document although he/she is not a user of the document. Therefore, project partners can keep the links to updated information all the way. Whether or when they will access the document is left up to them.

(2) Document Management

In SWIMS, project documents are annotated by metafiles. Conversely, the change of information in metafiles can be documented in the documents in a semi-automatic way. With the annotation, all documents are linked by two approaches: through the ontological concepts they "*are about*", and through their interdependencies, such as the "*refers to*", "*includes document*", "*depends on*", "*belongs to*", "*in conjunction with*", and "*is used to generate*" properties. All of the above attributes lead to such a result that SWIMS can complete many document management tasks that traditional information management systems cannot do. The most typical tasks are depicted in Figure 7-19, including *update document*, *classify document*, and *search information* in documents.

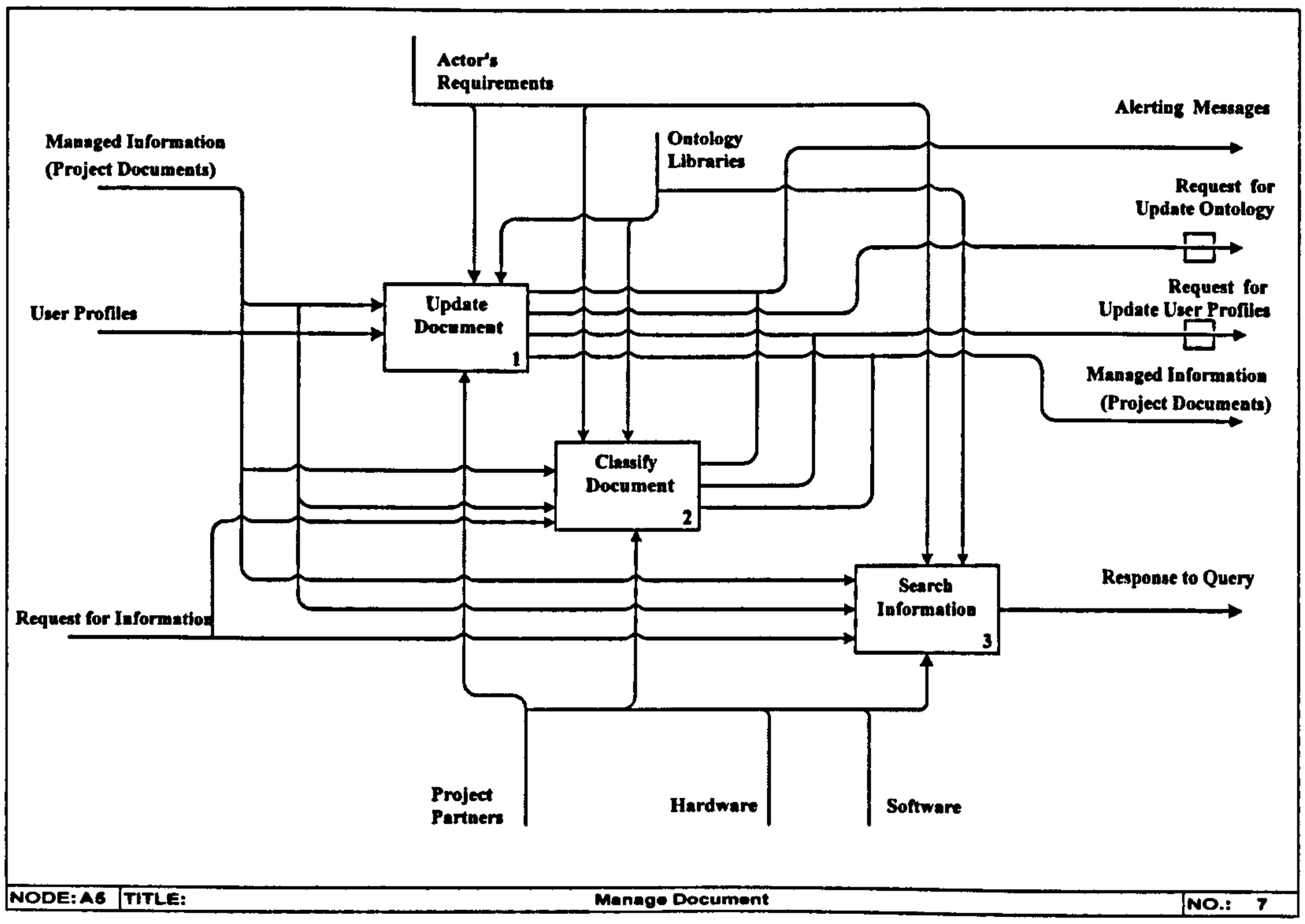


Figure 7-19: Document Management Process in IDEF0- Node A5

(i) Intelligent Document and Ontology Update

As mentioned before, documents are interdependent. Changes in Document A may lead to consequent changes in the documents if they "*refer to*" Document A. In this case, SWIMS Update Document module will identify the related documents and update them automatically. For documents that "*depend on*" Document A, SWIMS will send an update alerting message to the "authority" actor of those documents and ask them to identify if it is necessary to update the related documents.

Meanwhile, the change of information in documents may raise a request for modifying the SWIMS ontology. Project documents are annotated by a set of keywords and ontological concepts. Figure 7-20 illustrates the relationships between a document, a concept and their keywords. Within the figure, there are four relationships: *is about*, *is characterised by*, *has* and *is mate*.

"*Is about*" is a property of documents, which is the most direct link between the documents and the ontology. The values of "*is about*" property are instances of ontological concepts.

An indirect link between the documents and the ontology is via the "*is characterised by*" property. The values of "*is characterised by*" property are a set of keywords. Some of the keywords are inherited from "is about" concepts, because each concept "*has*" its own keywords, such as the Keywords 1, 2 and 3 in Figure 7-20. Other document keywords are granted by people or extracted from the document's body text, such as the Keywords A, B and C in Figure 7-20. When a concept and a keyword are used to annotate a document together, such as the "concept" and "Keyword C" in Figure 7-20, they are in a symmetric relationship: "*is mate*".

The interaction between documents and the ontology depends on the interaction between the values of "*is about*" and "*is characterised by*" properties. This has been addressed in Section 5.4.2-(2)-(ii). Here, their interaction is outlined from three perspectives: How the "*is about*" property defines the "*is characterised by*" property; how the "*is characterised by*" property affects the "*is about*" property; and how the "*is characterised by*" and "*is about*" properties affect the ontology.

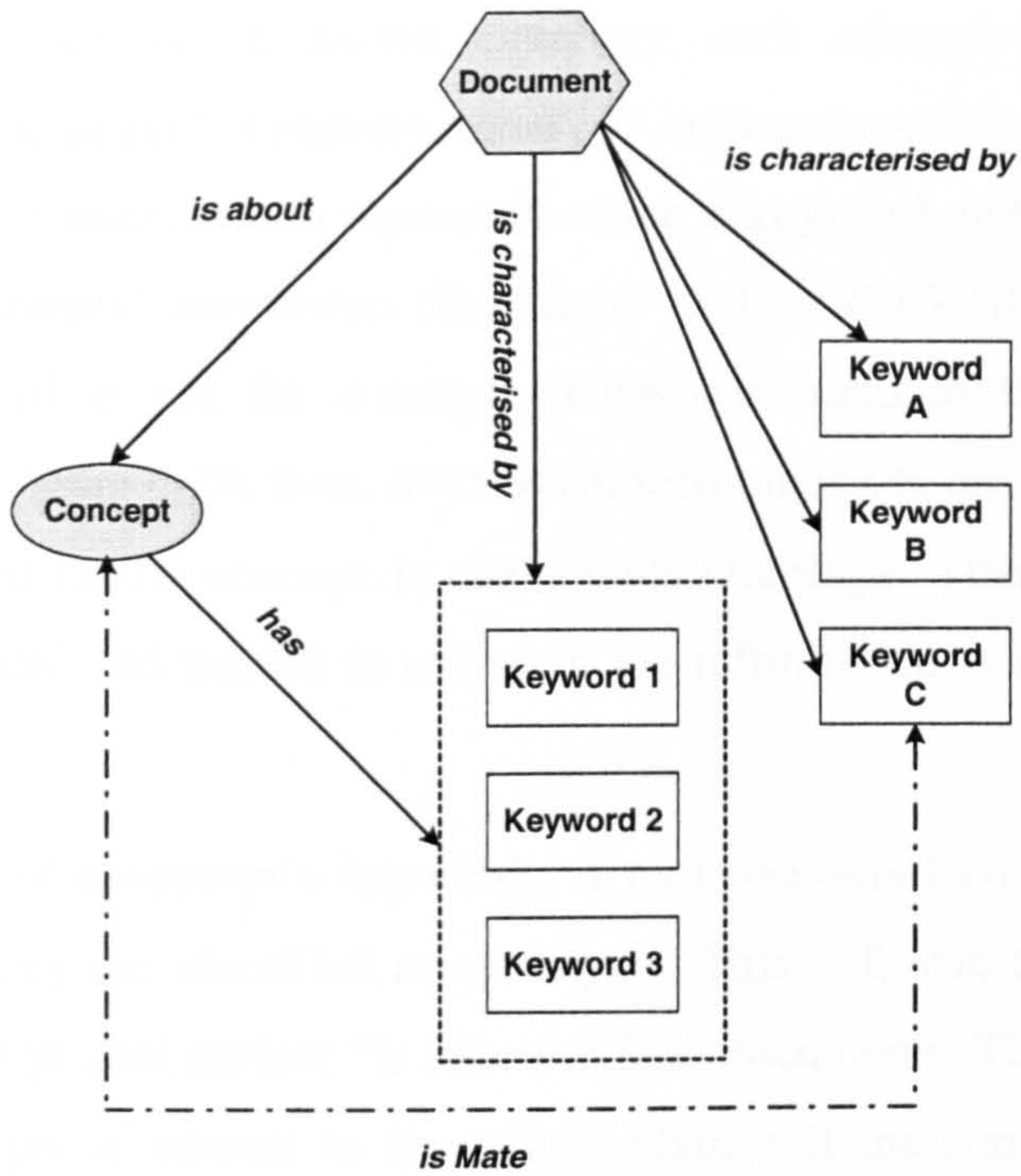


Figure 7-20: Relationship between Document, Concept and Keywords

How the "*is about*" property defines the "*is characterised by*" property: Concepts in the SWIMS Ontology are associated with a set of keywords. The keywords of the concepts, which a document "*is about*", are recommended to be included in the value of the document's "*is characterised by*" property, such as the Keywords 1, 2 and 3 in Figure 7-20.

How the "*is characterised by*" property affects the "*is about*" property: The "*is characterised by*" property indicates the document's keywords, such as the Keywords 1, 2, 3, A, B and C in Figure 7-20. These keywords can be specified by project partners, selected from the recommended terms of the "*is about*" concepts, or calculated from the document's body text by the vector model in Equations (5-1) to (5-3). If the people-specified keywords or computer-calculated keywords (such as Keywords A, B and C in Figure 7-20) match certain concept's keywords and this concept is not covered by "*is about*" property, then SWIMS will recommend to the project partners, who have "*authority*" over the document, to add this concept into the range of "*is about*" slot.

How the "*is characterised by*" and "*is about*" properties affect the ontology: Keywords and concepts are related in two circumstances: in the SWIMS Ontology or

in a document's metafile. In SWIMS Ontology, each concept *has* keywords. In metafiles, each document "*is characterised by*" keywords and "*is about*" concepts. If the frequency of mate, which represents when a keyword and a concept appear together in documents' annotation (Equations (5-4) and (5-5)), is high enough, while the keyword is not the concept's keywords, such as the "concept" and "Keyword C" in Figure 7-20, then, SWIMS will recommend to the project partners to map this keyword to the concept in the SWIMS Ontology. Therefore, the SWIMS Ontology is updated and trained to represent the information in project documents more accurately.

With the change of a concept's keywords, if two concepts have exactly the same keywords, then they are identified as synonyms. This will lead to a change in the user profile. Each project partner "*is interested in*" documents. The documents "*are about*" the concepts of interest to the project partner. If the concept has synonym concepts, then the documents that "*are about*" those synonyms will be brought into the partner's "*is interested in*" document list. This change can be calculated by OWL reasoners, such as the RacerPro.

(ii) Customised Classification and Computing Deductive Rules

The most traditional and basic document management method is the hierarchical category-based method. Currently, each project partner uses his own rules to categorise project documents and his/her own terms to name the folders. Because it is quite a personal or organisational convention, looking for related documents in a partners' computers becomes a complicated job for most people. Even to the person himself/herself, he/she may not remember the exact paths to a certain document if the document is related to more than one folder's subject and he/she kept many copies of the document in various folders. Thereby, if he/she wants to update the document, it is hard to ensure that all copies are updated and kept consistent across folders. The reason for the problem lies in the way information is managed. In traditional systems, information is managed at a document level and one document copy cannot exist in various folders. That conflicts with the nature of project information. The information is fragmented and a document involves multiple subjects. If a folder represents a subject, a document has to be copied into every related folder.

SWIMS can support end-users in customising the categorisation rules, and automatically classify documents. The individual document is not physically copied. SWIMS just links its metafile to the related categories. So when the information changed, its relevant documents in the information repository are updated and each document's associated metafile is updated. The updated metafile will be displayed in every related category. Therefore, the documented information in repositories can be shared and categorised at the same time. Thus consistency is kept throughout the information categories.

For example, there are 19 documents about the Royal Festival Hall Extension Building Project in the repository being shared between the client and project manager. All the documents are annotated by metafiles and the project partners are annotated by user profiles using SWIMS annotation model and ontology.

In the client's category, project documents are classified by their subject contents, including *administrative document*, *commercial document*, *legal document* and *technical document*. The category hierarchy is displayed on the left hand side of the window in Figure 7-21, and the project documents are listed in the middle of the window. The metafile of the selected document ("Acoustic Ceiling Specification) is displayed on the right hand side of the window. Figure 7-22 demonstrates the definition of folder technical document, which contains documents that are *in category of* technical. Where, technical is a sub-concept of subject content, which is a document related technique under technical topic. After classification by the OWL reasoner, RacerPro, the 19 documents have been categorised into four folders based on their subject topics. As illustrated in Figure 7-23, there are 14 technical documents, 3 legal documents, and 3 administrative documents. The document "Level Two Site Plan" is related to both "site instruction" (a type of technical document) and "machinery supply plan" (a type of administrative document). So it appears in two categories. The number of documents in each folder is marked in brackets. There are two numbers in the brackets next to each folder. The first number indicates the document number before classification and the number after the slash indicates the one after classification.

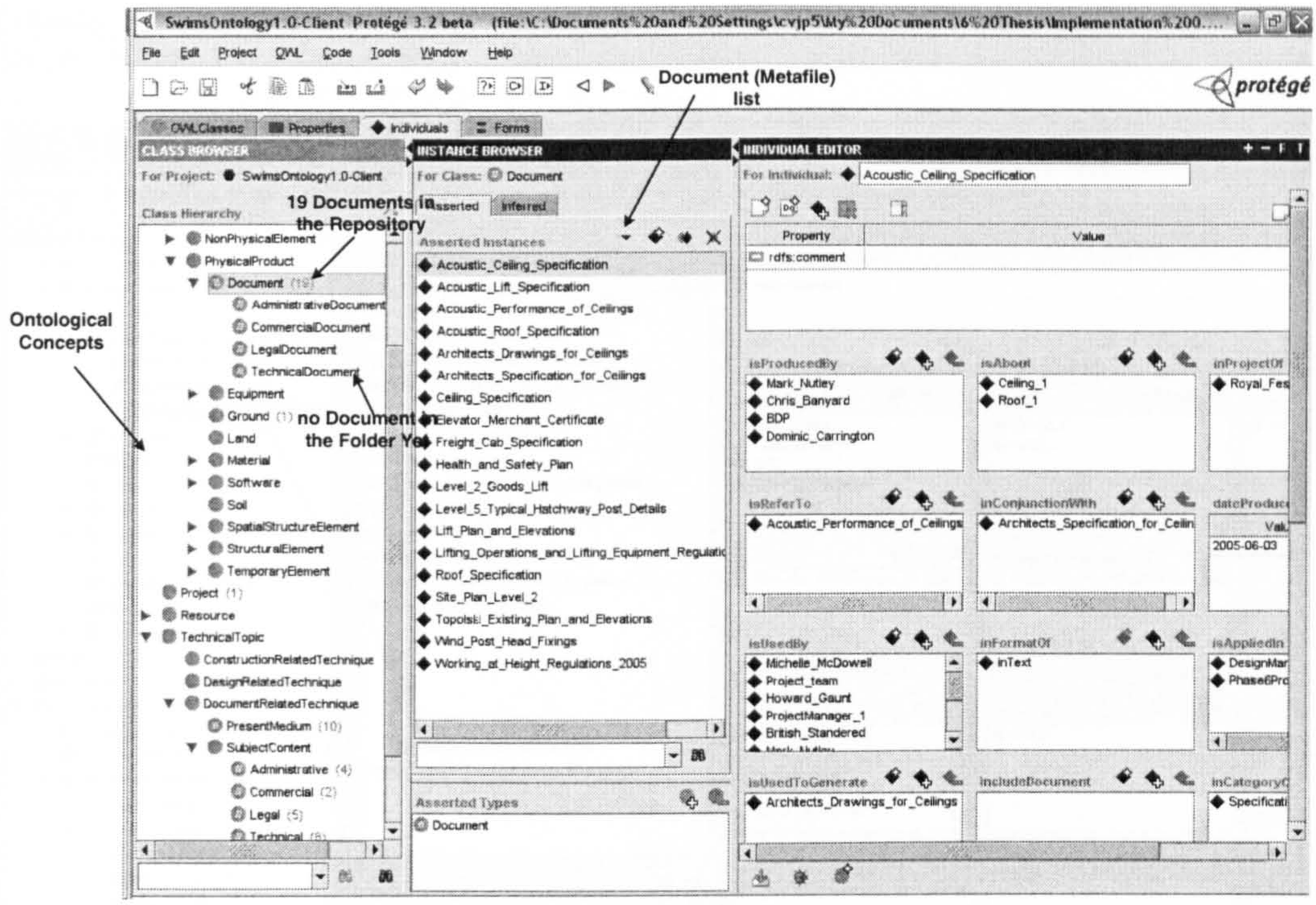


Figure 7-21: Client's Document Repository before Classification

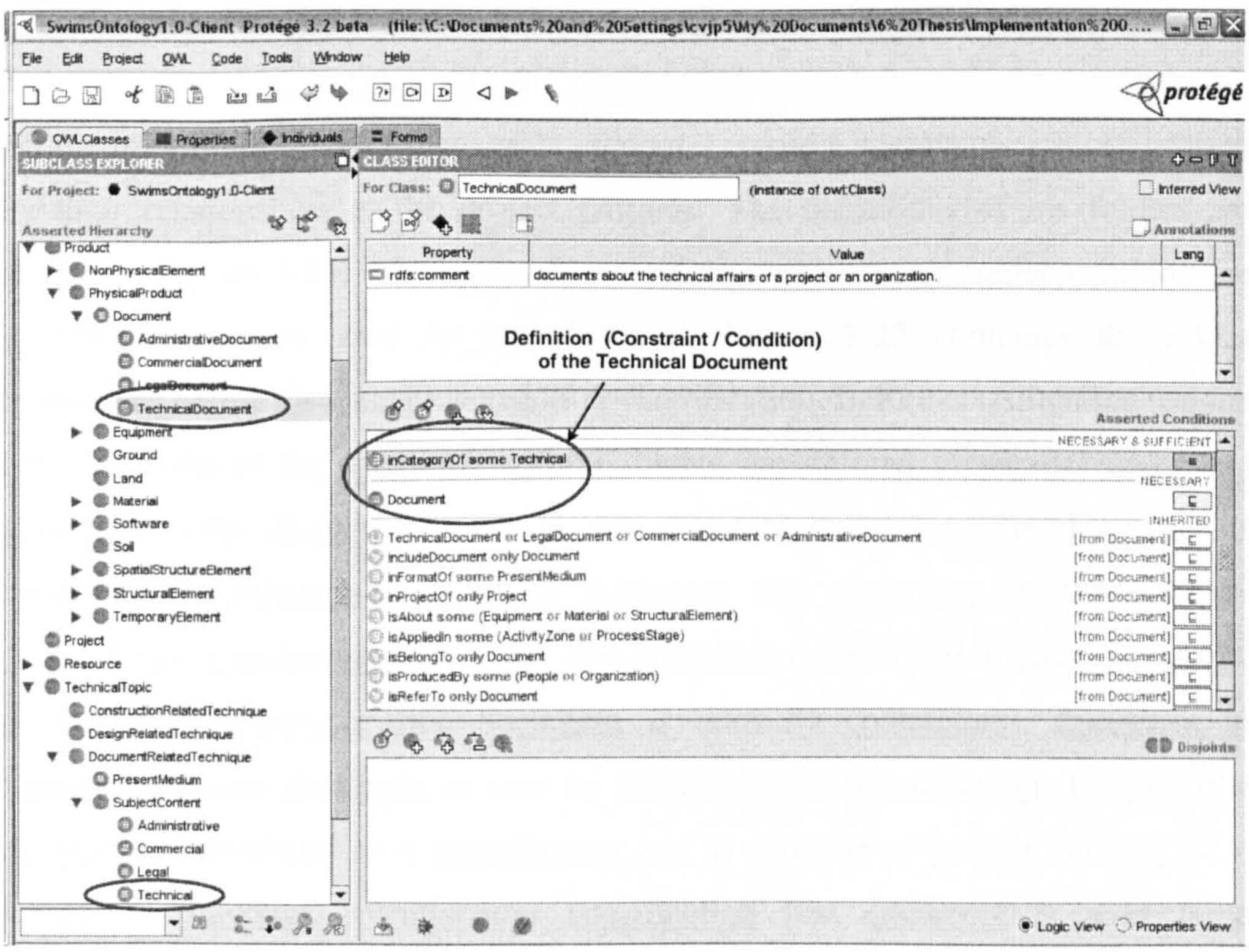


Figure 7-22: Definition of the Technical Document

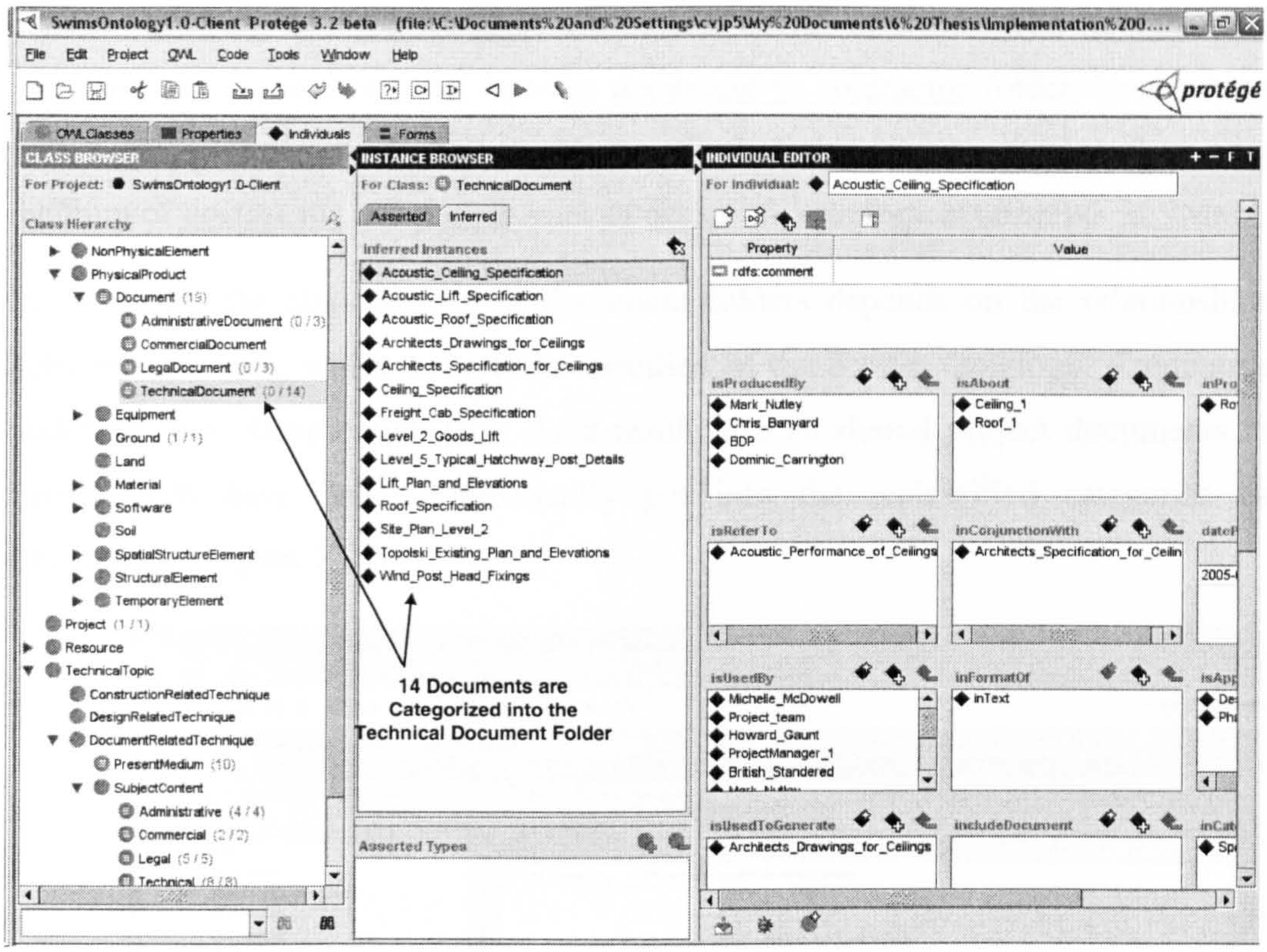


Figure 7-23: Classified Documents into the Client's Category

Meanwhile, in the project manager's computer, project documents are categorised by their relationships to the project partners. The definitions of his folders are shown in Figure 7-24, where, the document to manufacturer folder contains the documents that *are used by* manufacturers. Figure 7-25 compares the folder hierarchies before classification and after classification. Before classification (on the left hand side of the window), several folders are defined in parallel under the general concept document, such as the architect document (the document *is produced by* architects), contractor document (the document *is produced by* contractors), manufacturer document (the document *is produced by* manufacturers), document to contractor (the document *is used by* contractors), document to manufacturer (the document *is used by* manufacturer), manufacturer drawing (the document *is produced by* a manufacturer and *in category of* project drawing or *in format of* drawing), manufacturer specification (the document *is used by* a manufacturer and *in category of* specification), etc. These folders are reclassified by the OWL reasoner as shown in the middle of the windows in Figure 7-25. The architect drawing folder is moved into the architect document folder and the architect specification folder is moved into the document to architect folders,

because drawing and specification are certain document types. The document to manufacturer folder is moved into the document to contractor folder, because the manufacturer is a sub-class-of contractor. Both the subject content and present medium of documents and people roles of actors are ontological concepts in SWIMS.

In summary, the classification of document folders depends on the relationships between concepts, which have been specified in the SWIMS Ontology. Computers undertake the classification task. As a result, the 19 shared project documents in Figure 7-26 have been automatically put into the reclassified categories as illustrated in Figure 7-27.

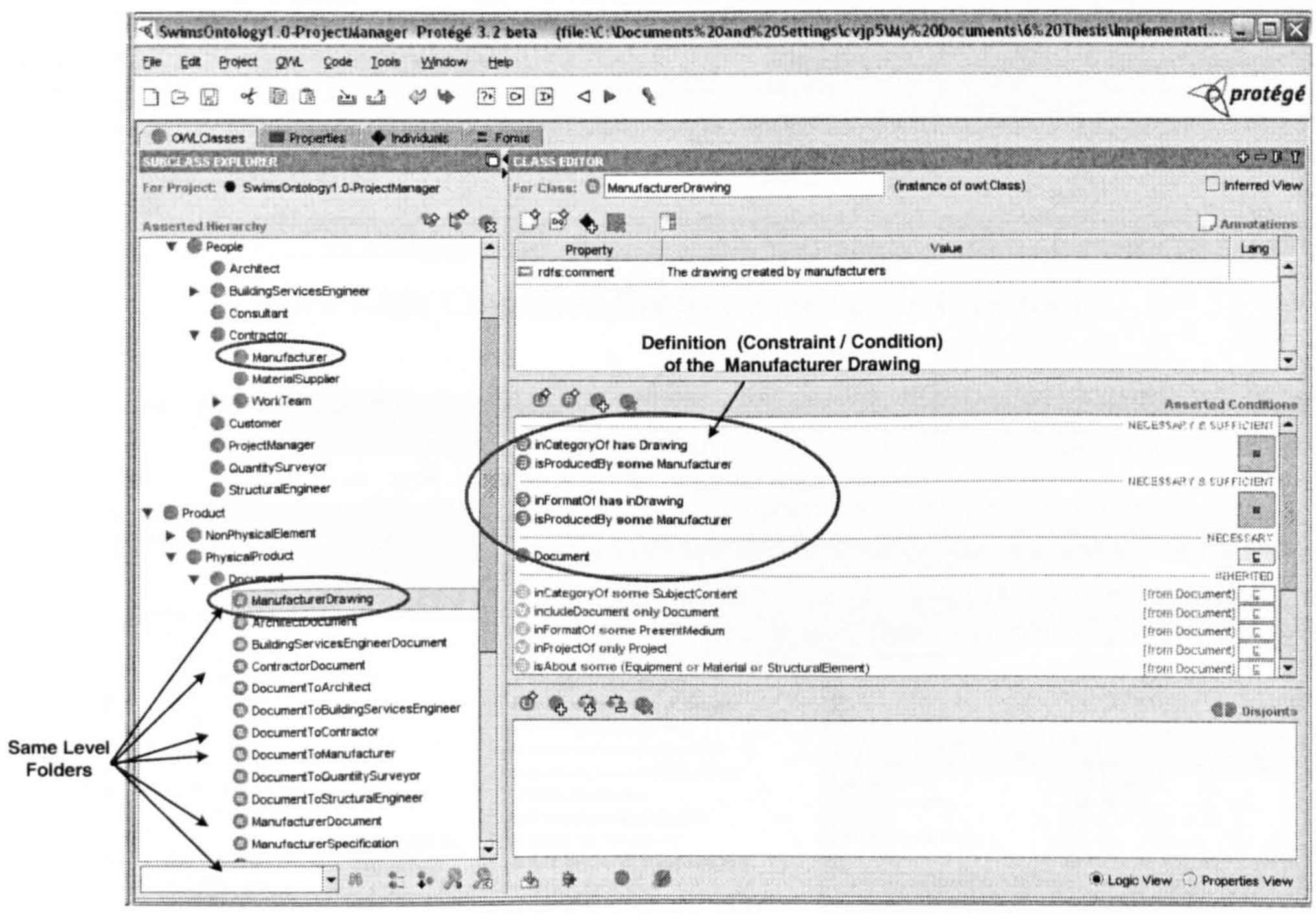


Figure 7-24: Definition of the Manufacturer Drawing

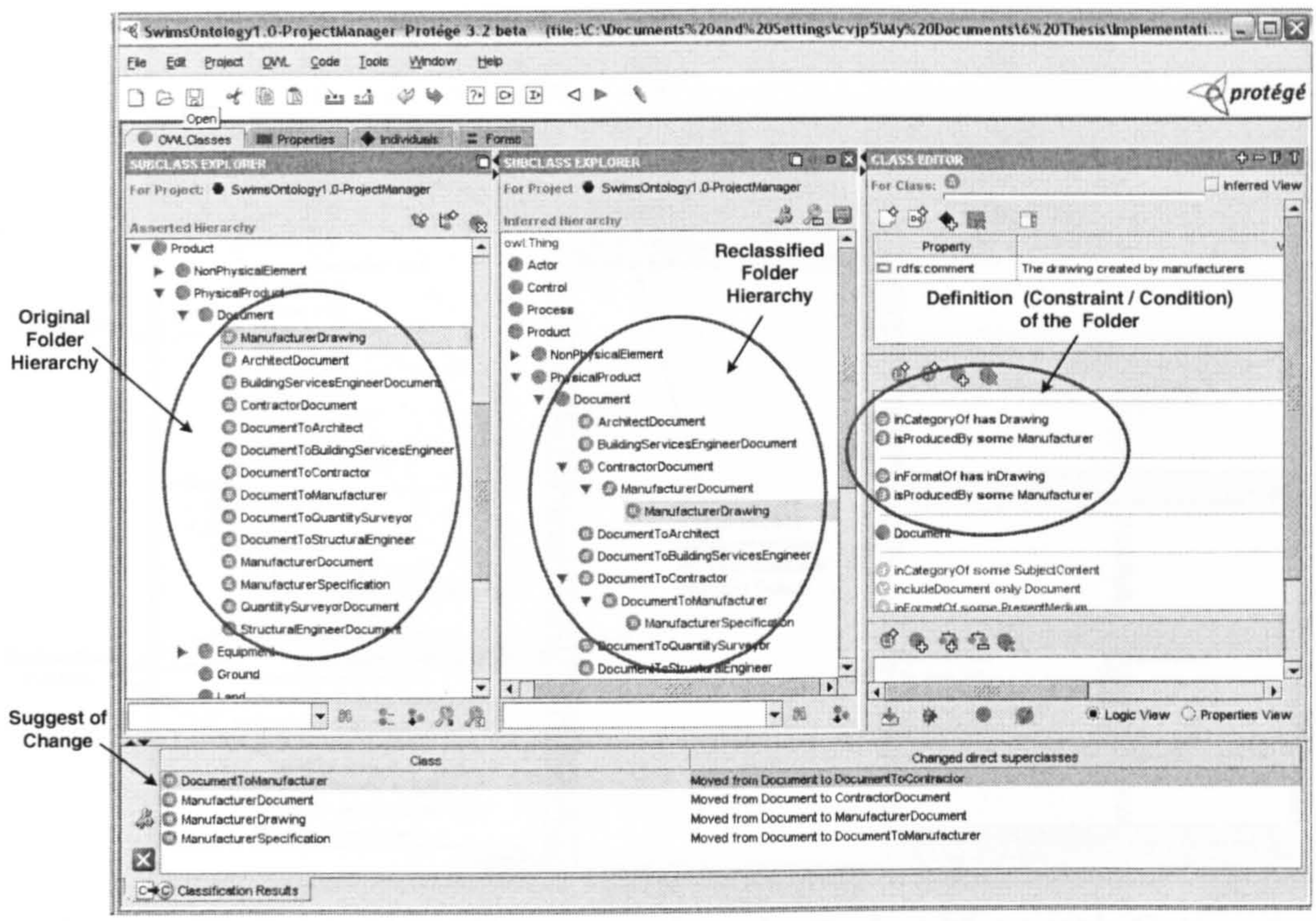


Figure 7-25: Classified Project Manager's Category

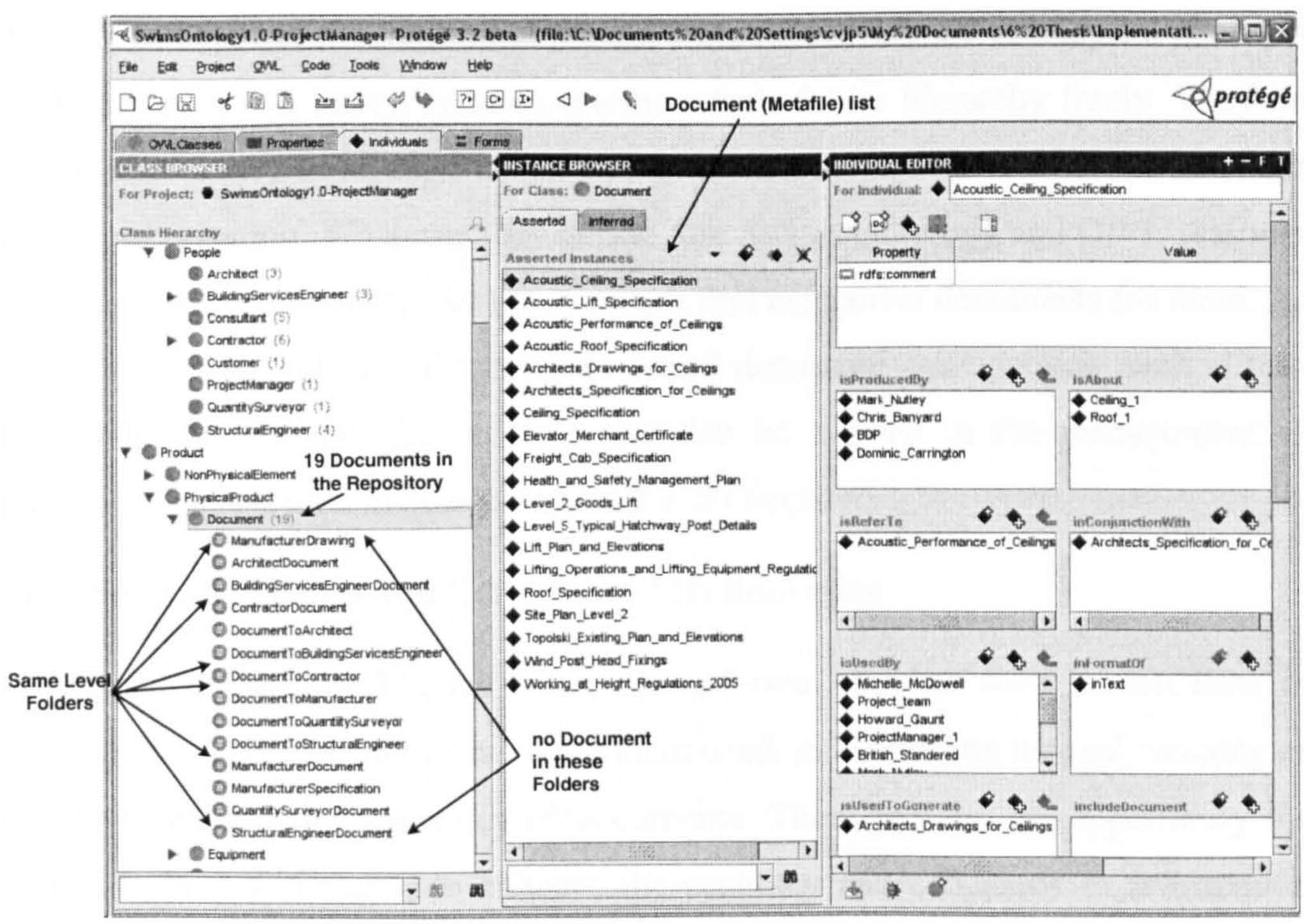


Figure 7-26: Project Manager's Document Repository before Classification

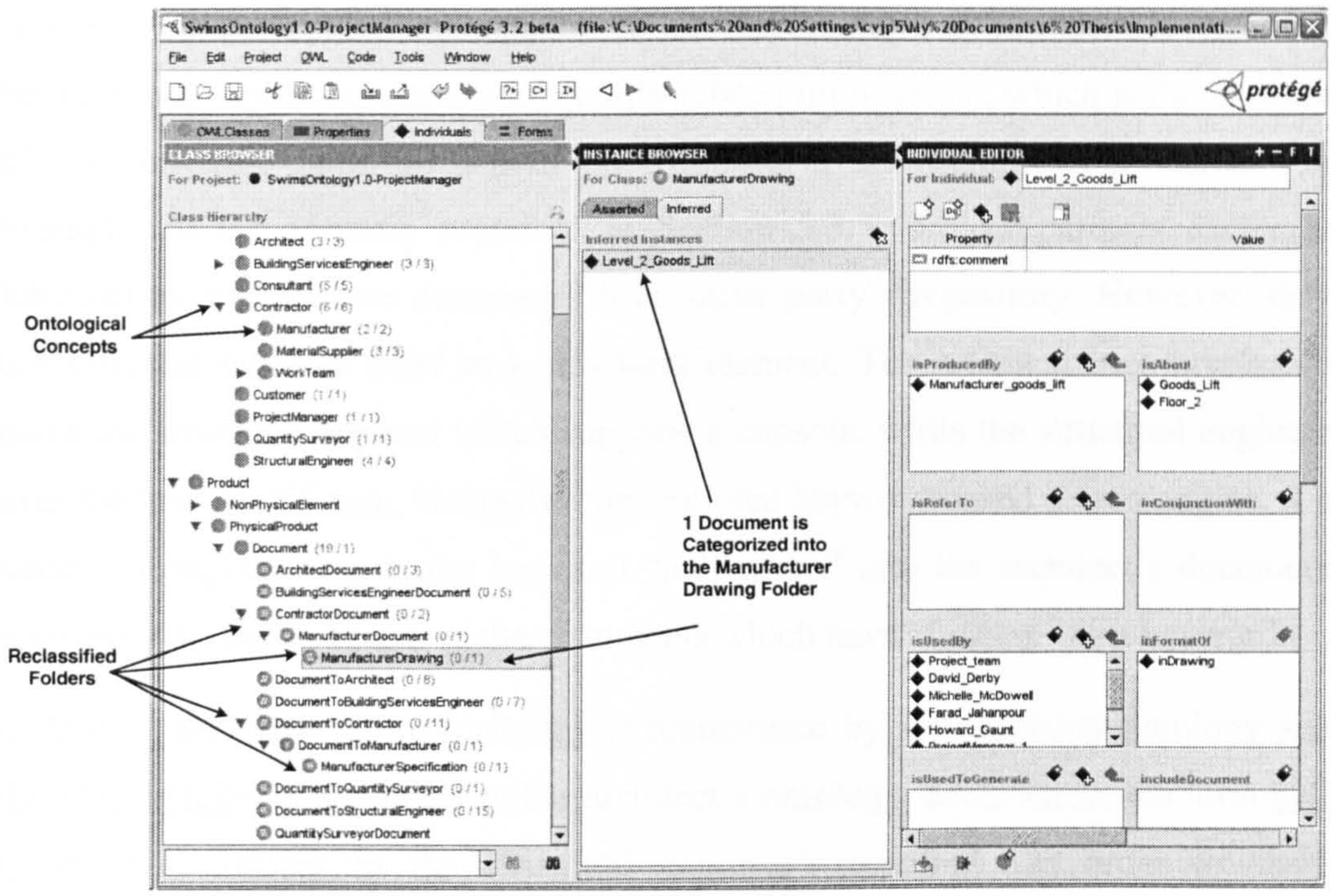


Figure 7-27: Categorising Documents into the Project Manager's Category

The above example illustrates that project partners can use the ontological concepts to define their own category and customise their folder hierarchy freely. They do not need to worry about how to keep their documents updated and consistent after their customisation. Whatever they define, the SWIMS Ontology and OWL reasoner can check the consistency, classify concepts and categorise documents for them. As a result, SWIMS improves the category-based document management method to a knowledge-based one. This method can also be applied in the management of human resources, such as classifying staff and clients.

(iii) Semantic Search and Converting Terminologies

As argued in Section 3.4, current keyword-based searches save human time in looking for electronic documents but remain weak in identifying the real meaning of the keyword and the semantics of documents. There remains an opportunity for Semantic Web techniques to enhance the accuracy and efficiency of information retrieval. Since the information of project partners and the content of documents have been annotated in user profiles and metafiles, a Semantic Web-based search can achieve the desired effects. It can address the real demands of project partners by identifying their job responsibilities and personal interests. It can also access the

content and context of documents. More importantly, it can deduce relations between ontological concepts and identify related information, which is the instance of the concepts. Thus the search on SWIMS is a knowledge-based search. For example, in the scenario proposed in Section 5.3.3, project partners exchange information and retrieve documents from other party's repository. However, they use different terms to refer to a structural element. The architect uses bracket to name the structural element which supports a console, while the structural engineer uses the term Half Beam. Using the conventional keyword-based search engine, if a structural engineer inputs the keyword "half beam" into the architect's document repository, he can not retrieve the documents which have "bracket" as a keyword.

In SWIMS, the architect's vocabulary is represented by his speciality ontology and the term bracket is a concept in the architect's ontology. Meanwhile, the term half beam is a concept in the structural engineer's ontology. In order to share information, SWIMS maps related concepts and merges the two ontologies. This task was conducted using PROMPT in Protégé, which involved three steps:

Step 1: Compare two ontologies (as shown in Figure 7-28);

Step 2: Aggregate concepts, properties, and instances from two ontologies and save them in a temporary ontology (as shown in Figure 7-30);

Step 3: Deduce relationships between concepts and map related concepts (as shown in Figure 7-32).

The architect's and structural engineer's ontologies are aggregated into the one displayed on the left hand side of the window in Figure 7-29.

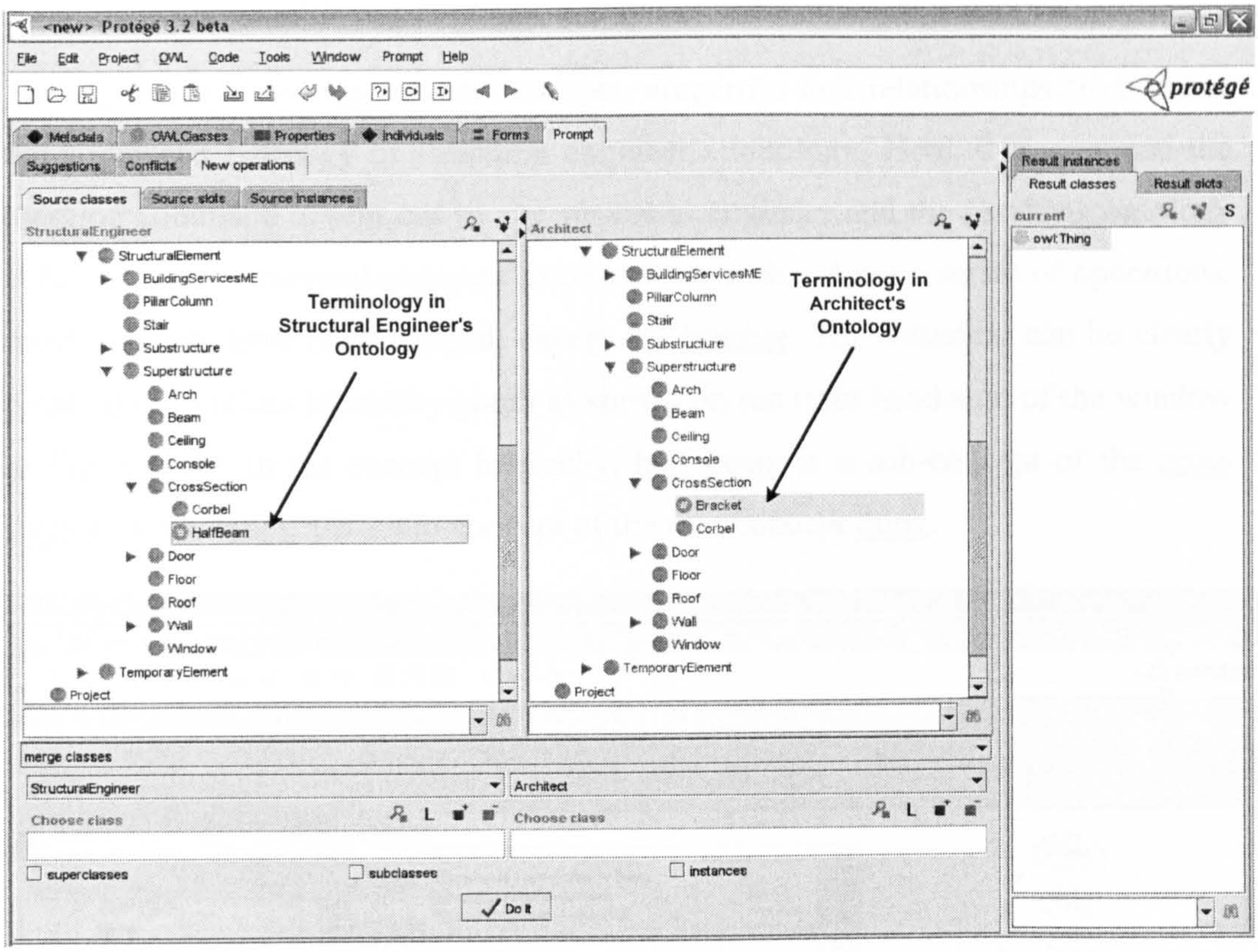


Figure 7-28: Compare Terminologies of Architect and Structural Engineer

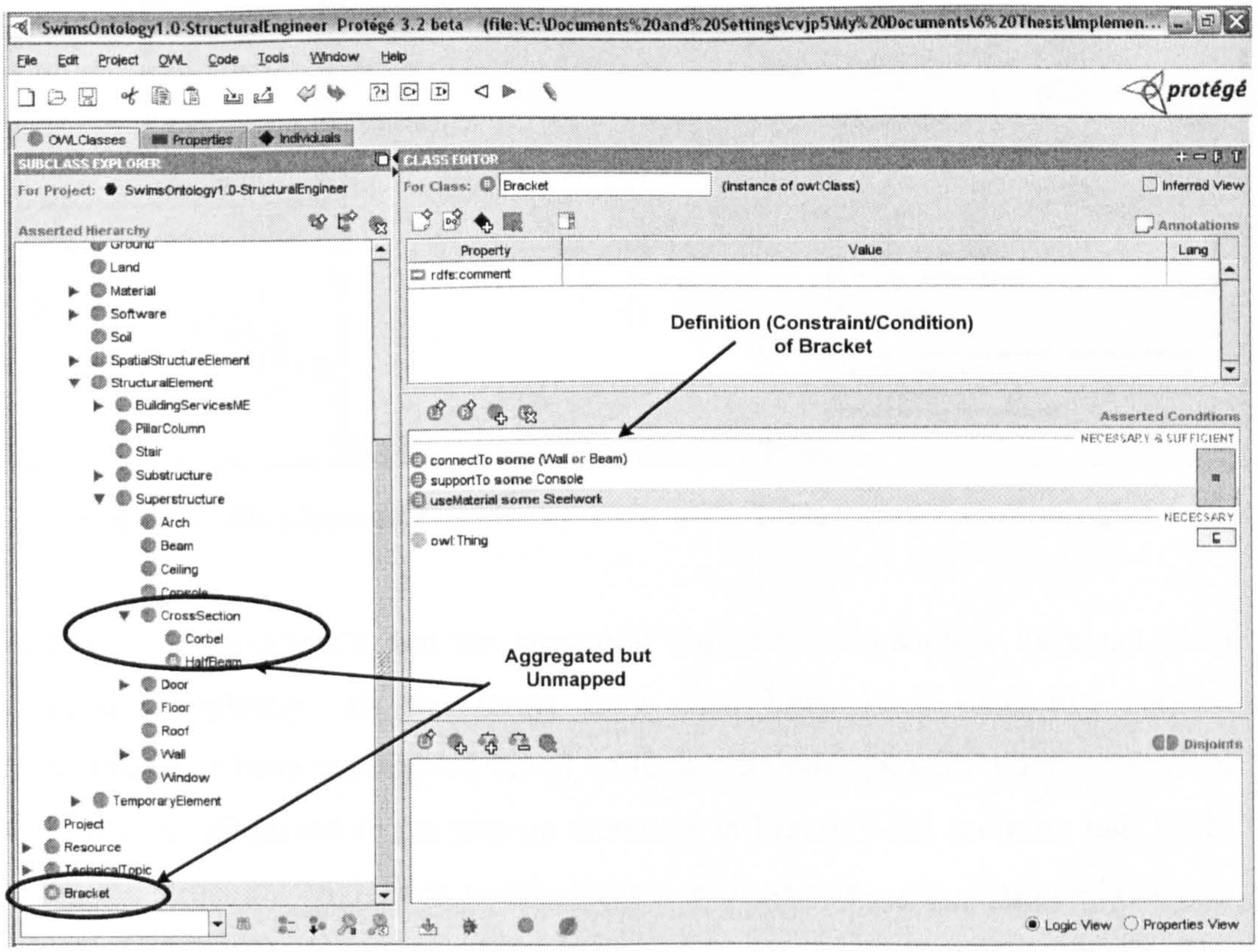


Figure 7-29: Aggregated Terminologies before Mapping

The aggregated ontology covers concepts, properties and relationships from either the architect's ontology or structural engineer's ontology. Here, it is assumed the merging command is sent out by the structural engineer and the resulting ontology is based on the structural engineer's concept hierarchy. After a series of operations, most concepts have been merged, except the bracket. This situation can be clearly pictured in the class hierarchy graph as shown on the right hand side of the window in Figure 7-30. In the concept hierarchy, half beam is a sub-concept of the cross section, while bracket is a sub-concept of the root concept thing.

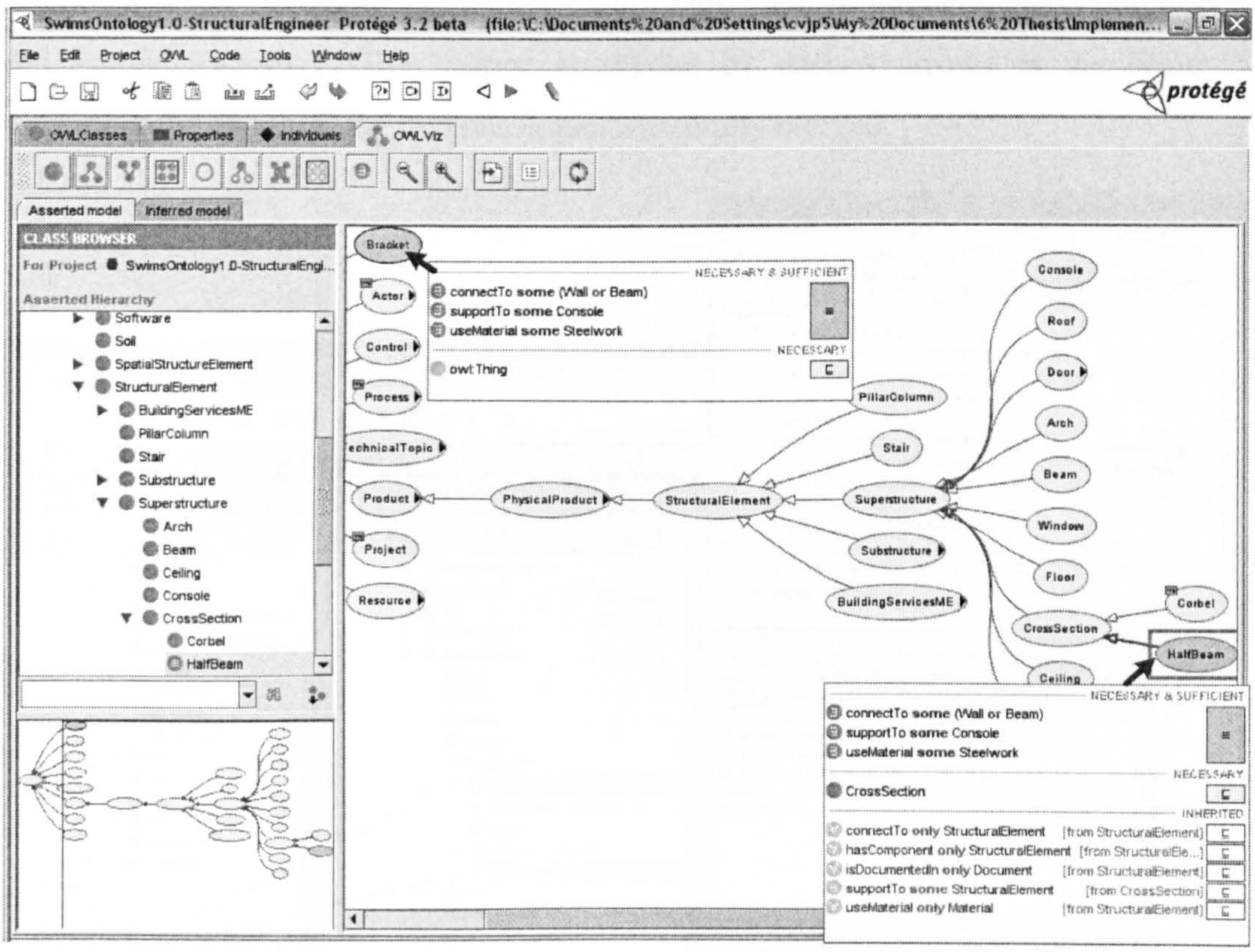


Figure 7-30: Concept Hierarchy View with Unmapped Terminologies

Although the architect's and the structural engineer's vocabulary have not been mapped completely, all the terms they used have been explicitly defined (constrained or have conditions) based on their relationships with other ontological concepts. As displayed in the pop-up windows in Figure 7-30, the term half beam (from the structural engineer's vocabulary) is defined as a cross section which *uses material* of steelwork, *connects to* a wall or a beam, and *supports* a console. The architect uses the term bracket for the support of the console. Bracket is defined as a

direct concept under the root concept in the aggregated ontology. It uses *material of* steelwork, *connects to* a wall or a beam and provides *support to* a console. The OWL reasoner is then used to deduce the relationship between the terms bracket and half beam. In this case, because the concepts: wall, console, beam and steelwork have been merged, and bracket and half beam are defined by these merged concepts, the reasoner deduces the defined rules, and induces that the term bracket is a synonym of half beam. Figure 7-31 displays the unmapped ontology (aggregated ontology) on the left hand side of the window, and the inferred ontology (merged ontology) next to it, in the middle of the window. Recommended changes are displayed at the bottom of the window and highlighted in the aggregated ontology and merged ontology. The change is market by dashed arrows in the figure. Accepting the changes, the two ontologies are totally merged.

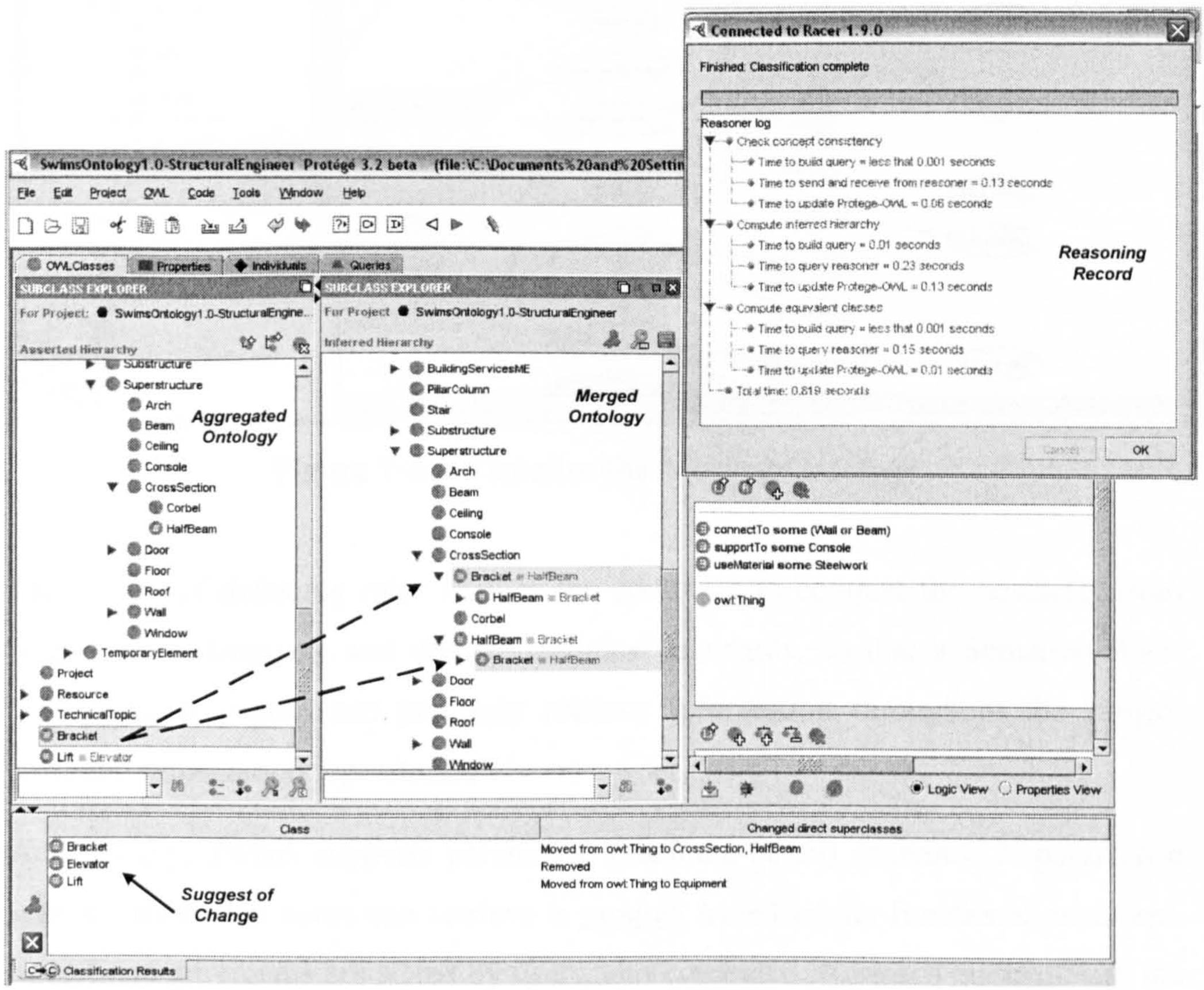


Figure 7-31: Mapping Terminologies and Merging Ontologies in Protégé

As a result, the merged ontology was reclassified, as illustrated in Figure 7-32. The concept bracket is linked to half beam by the *same as* relationship and integrated

into the concept hierarchy. The instances of bracket and half beam are virtually merged as well after reasoning. The documents, which *are about* the bracket in the architect's document repository, are accessible via either the half beam or bracket concept using the merged ontology.

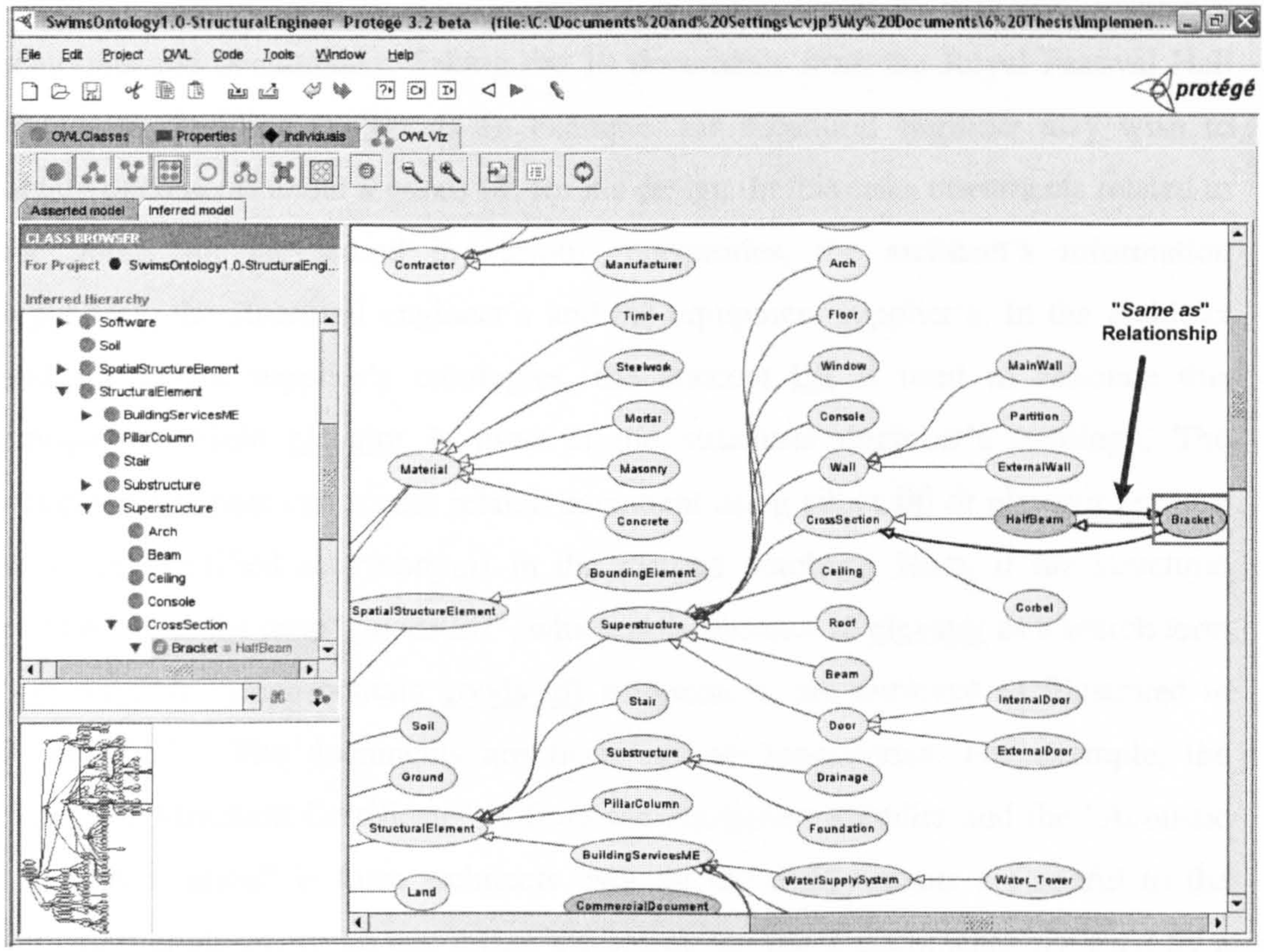


Figure 7-32: Visualise the Merged Ontology

The ability of deducing rules enables the computer to conquer the obstacles from jargon, multi-language and diverse technical standards, so that a Semantic Web-based search engine can precisely retrieve information throughout the project partners' information repositories.

Additionally, SWIMS supports parametric (attribute-based) searches. A parametric search means that users can retrieve a product based on its functional attributes. Specific requirements are stated by users, and compared by search engines with the attributes of products to determine appropriate matches. For example, a client can search for a structural element with a specified shape, material, and mechanical parameters. Within the SWIMS documented information management scenario, project partners can search a document by its semantic properties/attributes, as

shown in Figures 8-33 to 8-39. The Semantic Web-based parametric search has an advantage over a traditional attribute-based search. The terms used to describe product attributes are based on a traditional attribute-based search. However, with SWIMS, synonyms, terminologies, and even technical standards are convertible, all of which can be used to retrieve related products (such as structural elements, materials and documents). Taking the 19 documents from the Royal Festival Hall Extension Building Project as an example, the structural engineer may wish to obtain documents about a goods lift for his design. In this case, documents related to the goods lift are stored in various repositories, the architect's information repository, the structural engineer's and the equipment supplier's. In the architect and equipment supplier's ontologies, the concept lift is used to annotate this equipment, while elevator is used in the structural engineer's ontology. The structural engineer can access related document using either lift or elevator because they are identified as synonyms in the merged ontology. Here, if the structural engineer uses the term "goods lift", which is an instance of elevator as a search term, 8 documents, which contain goods_lift information, are retrieved as illustrated in Figure 7-33. The documents are from various repositories. For example, the "Elevator Merchant Certificate" is from the equipment supplier and the "Acoustic Lift Specification" is from architects. Not all the 8 documents are useful to the structural engineer, so he can refine the search as: those documents *are used by* a structural_engineer, so that he does not need to manually filter out the document irrelevant to him. Then, the results are narrowed down to 4 documents (Figure 7-34). Within these, he can search for those used in the full_conceptual_design stage. This results in only 3 documents (Figure 7-35). The information in the documents is accessible through the use of metafiles.

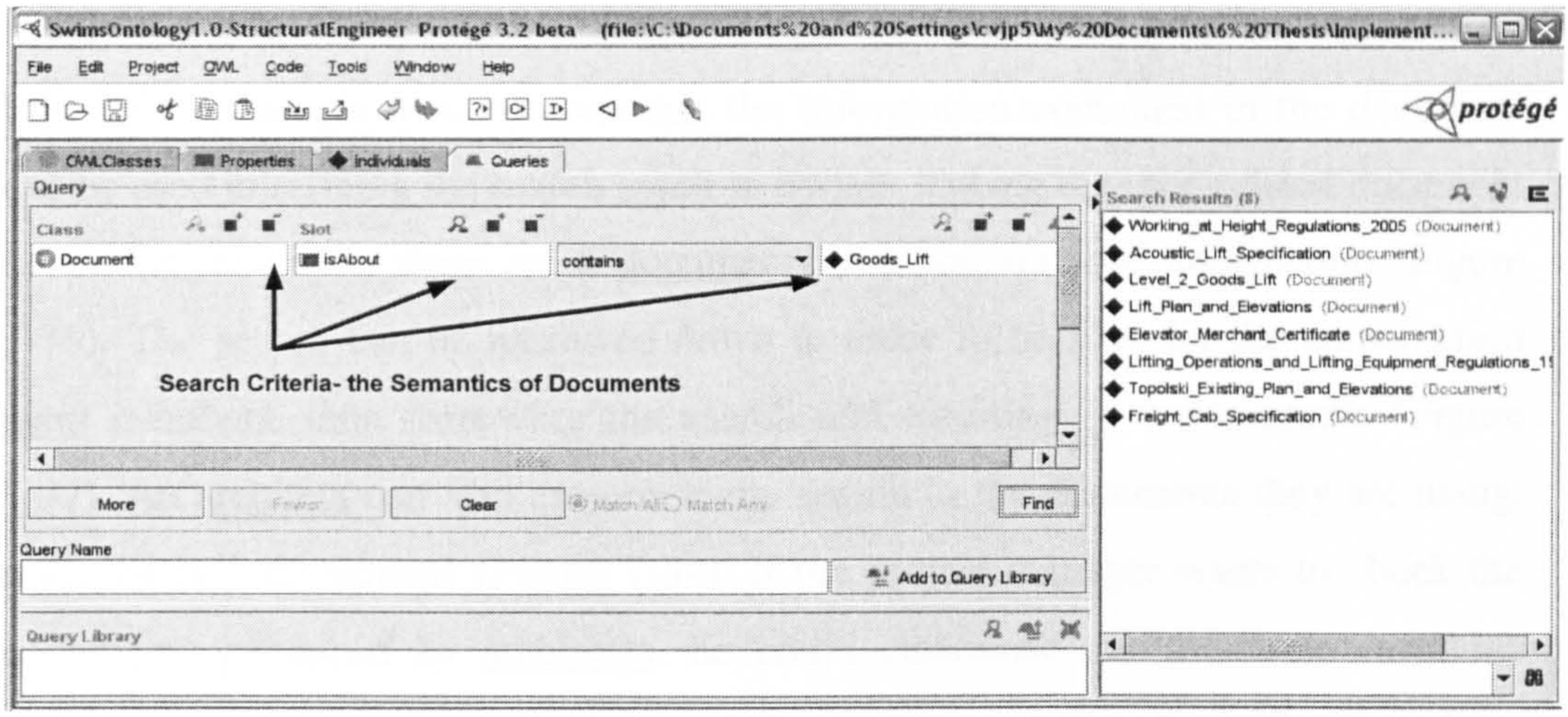


Figure 7-33: Search Documents Based on Their Semantics-Example 1a

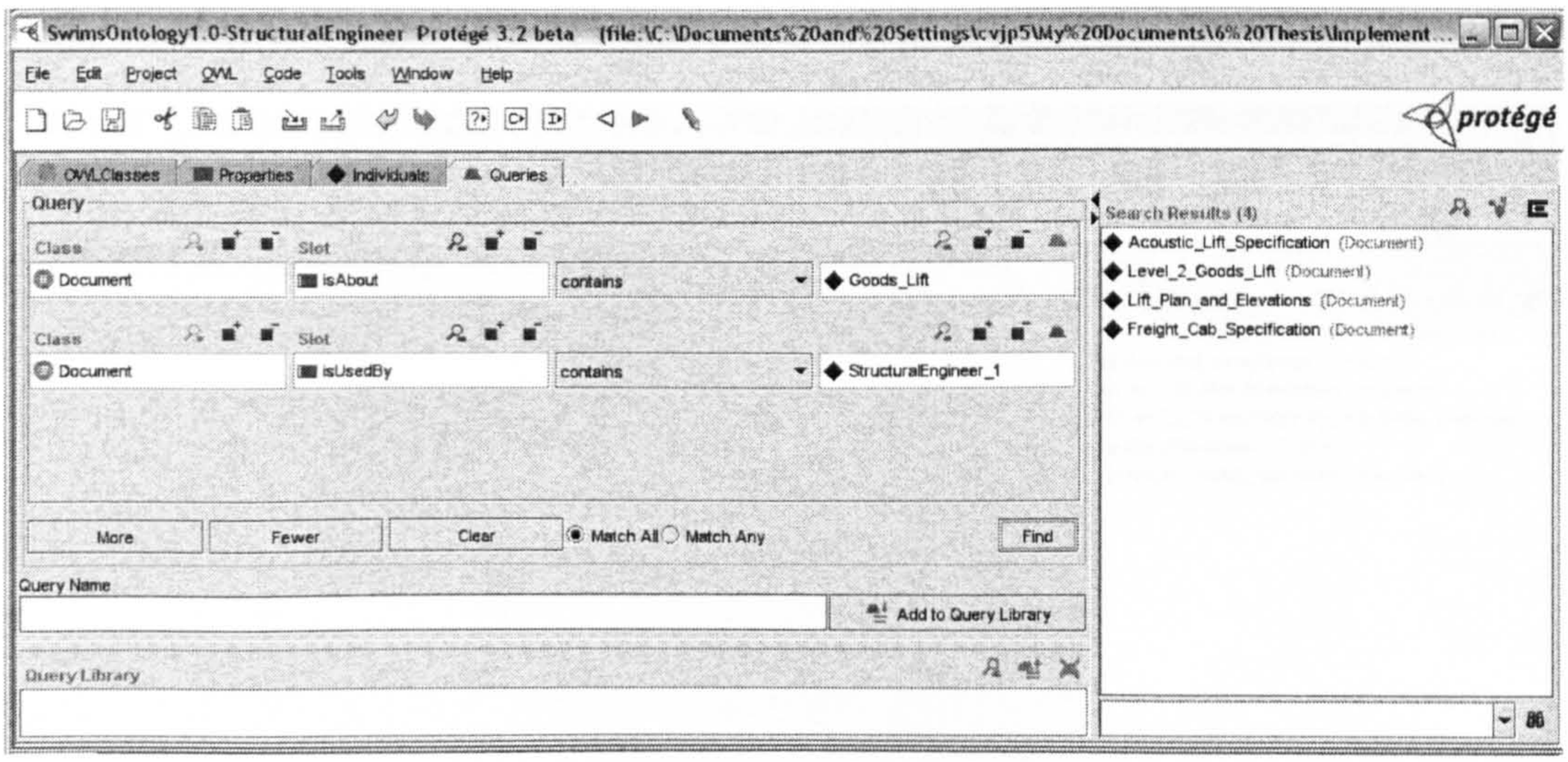


Figure 7-34: Search Documents Based on Their Semantics-Example 1b

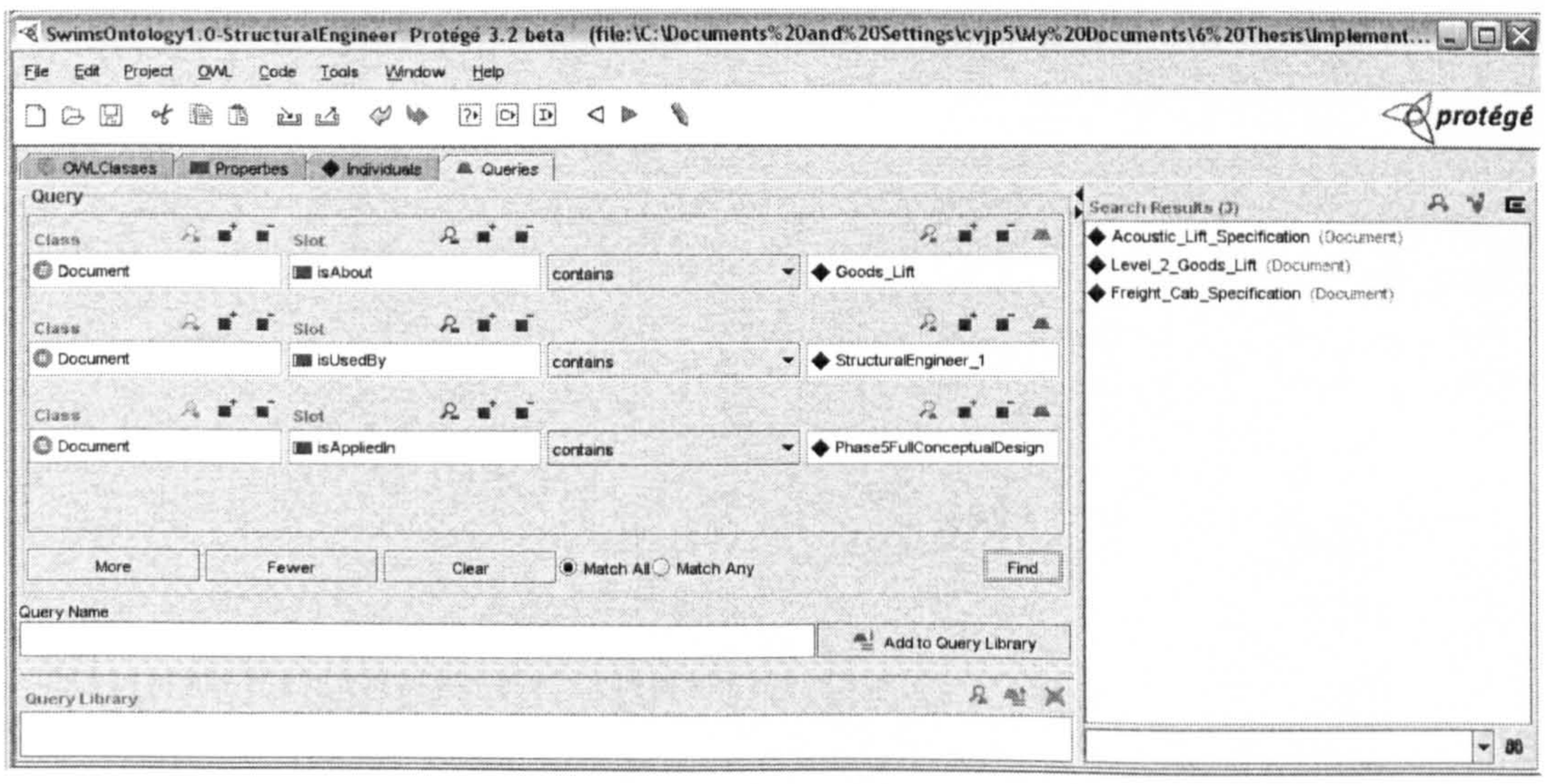


Figure 7-35: Search Documents Based on Their Semantics-Example 1c

The above example demonstrates that the information contained in the document can be used in refining the search result in SWIMS. Taking the roof-related document as another example, there are 6 documents that contain roof information (Figure 7-36). The search can be narrowed down to those to be used by the construction team members, thus narrowing the search and resulting in 4 documents (Figure 7-37). An architect can also customise the search to the documents they are using, resulting in 3 documents (Figure 7-38). If the project manager wants to check the documents *produced by* architects, he could customise the search to meet his requirements. This reduces the relevant documents to 2 (Figure 7-39). Therefore, project partners are able to access the most appropriate documents using a Semantic Web-based search engine.

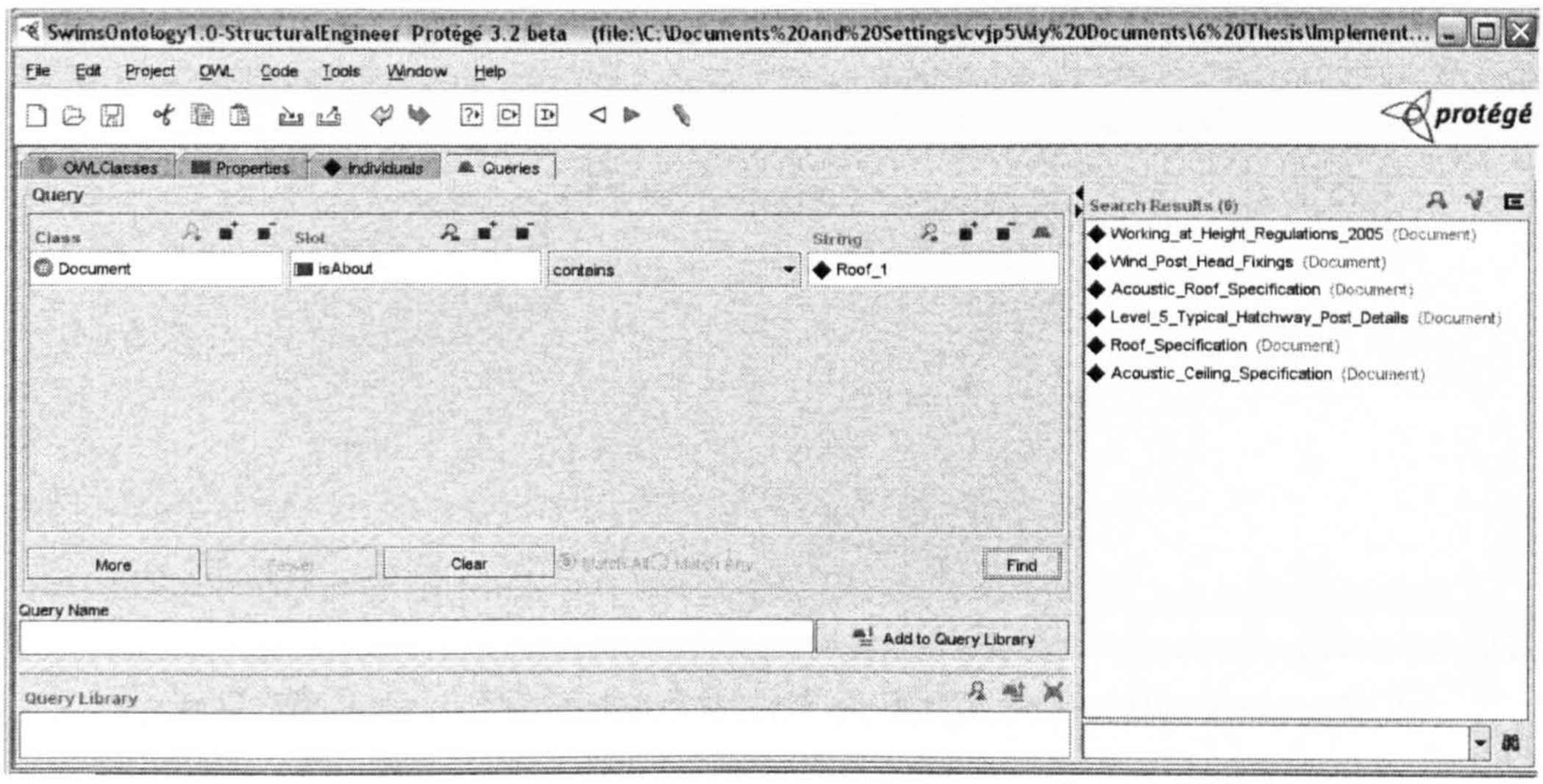


Figure 7-36: Search Documents Based on Their Semantics-Example 2a

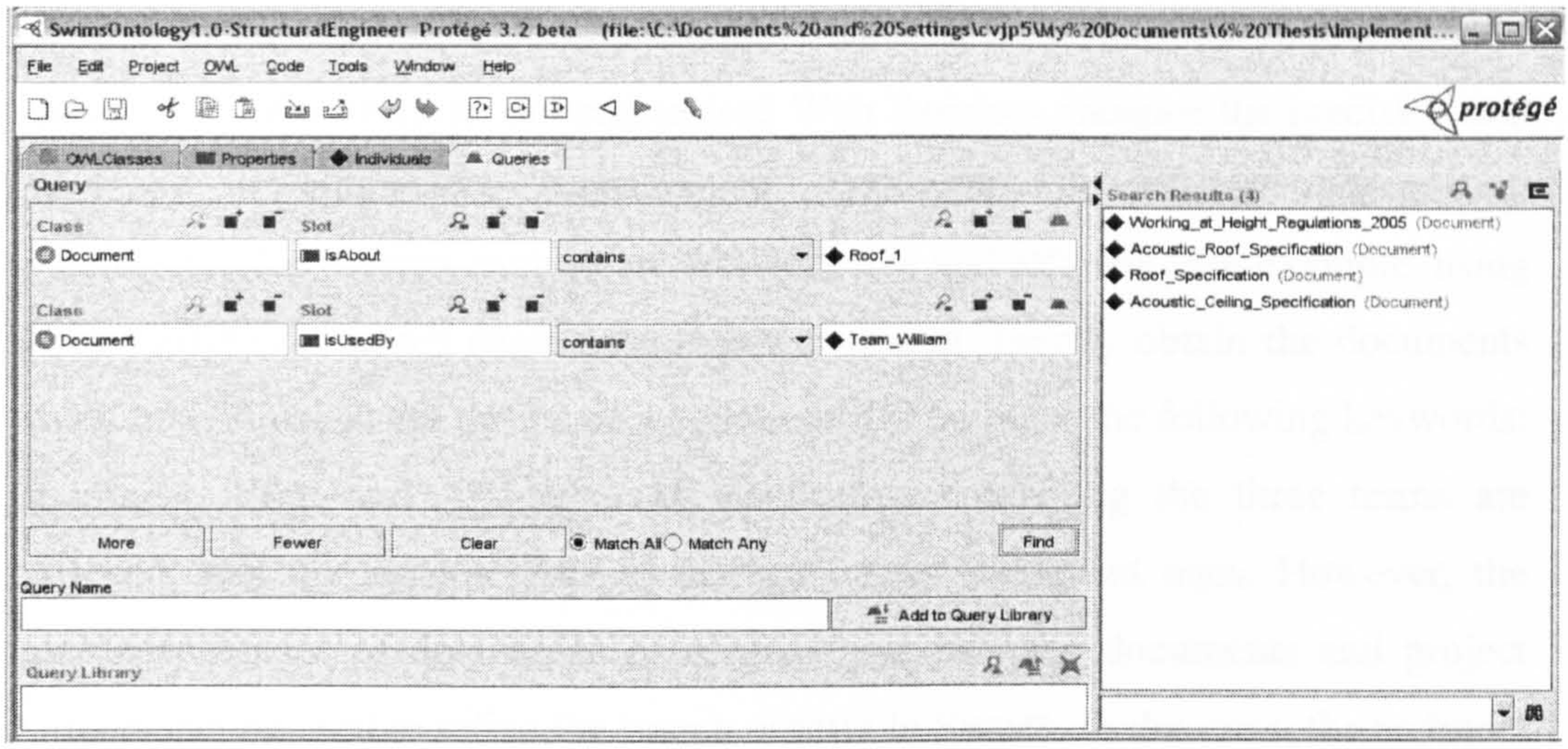


Figure 7-37: Search Documents Based on Their Semantics-Example 2b

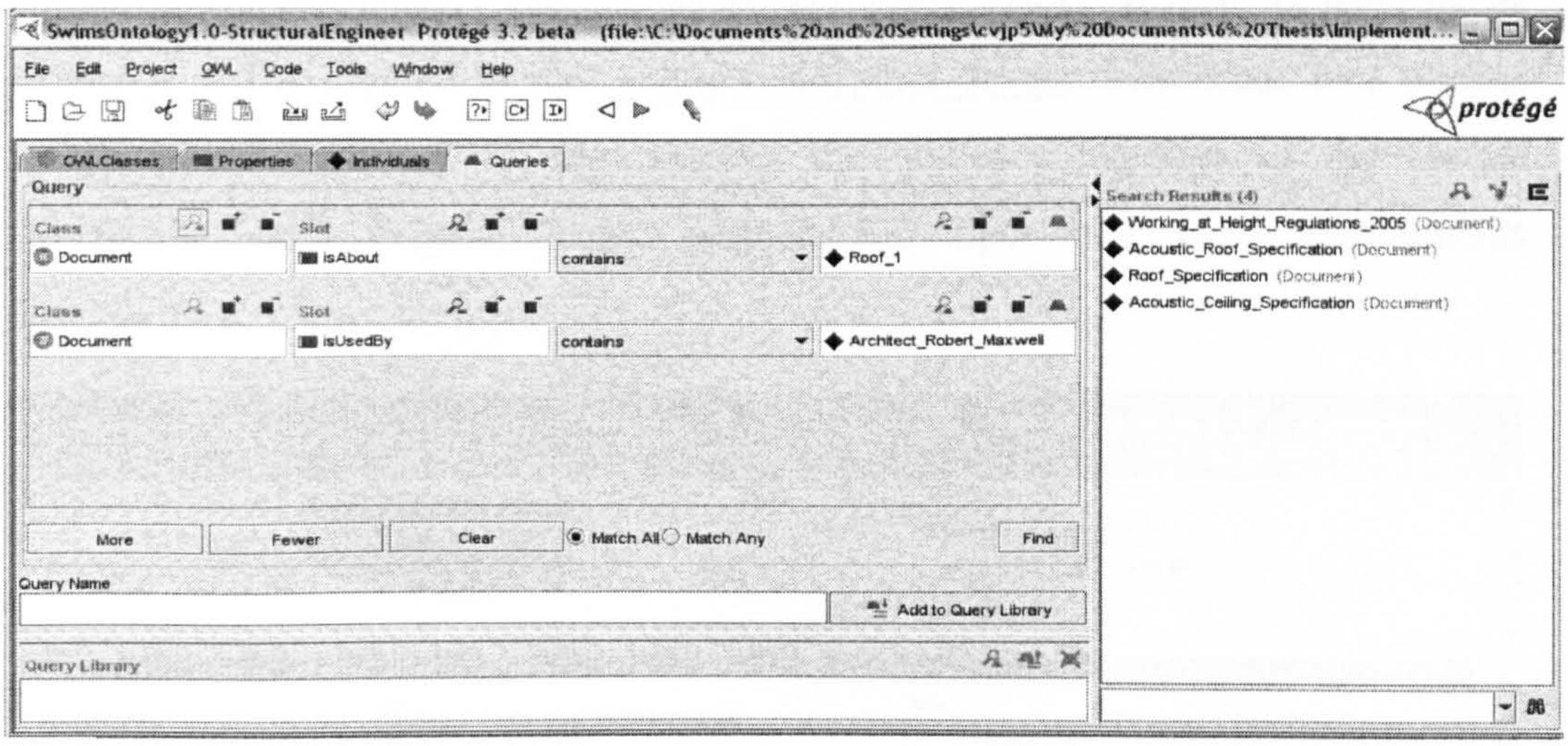


Figure 7-38: Search Documents Based on Their Semantics-Example 2c

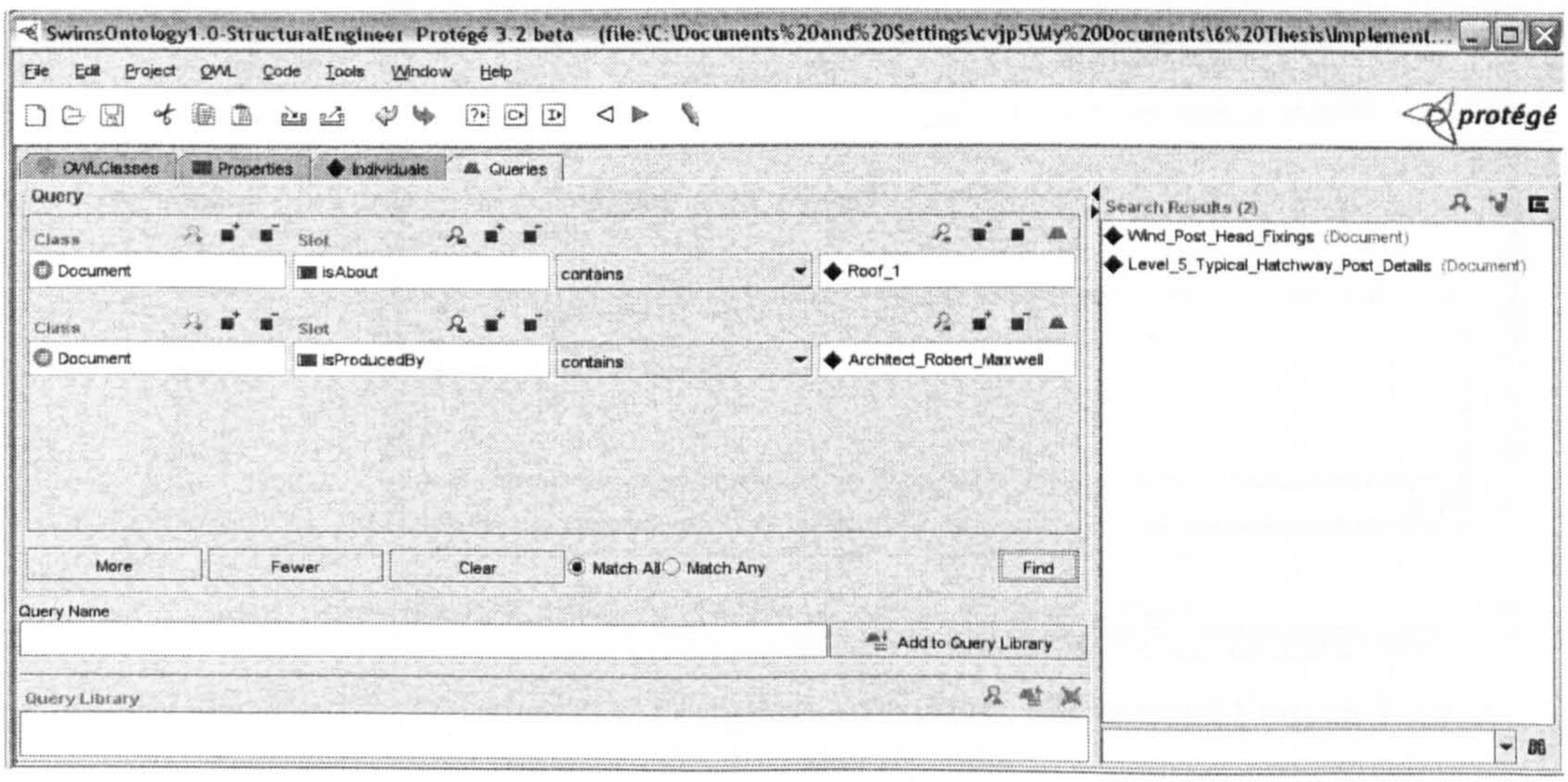


Figure 7-39: Search Documents Based on Their Semantics-Example 2d

These information-centric and customised Web Services enhance the precision and efficiency of information management. Traditional information management systems and Web search engines are less effective and efficient. For example, using a keyword-based search engine, an architect cannot directly obtain the documents which he will use in the design of a certain roof if he input the following keywords: Document, Roof and Architect. All documents containing the three terms are retrieved, and the architect has to manually filter irrelevant ones. However, the interdependencies of documents or relationships between documents and project partners can be used to refine the search criteria in SWIMS. In this case, the architect can specify the relationship between the document, roof, himself and his search objective. He can specify the search as: document *refers to* roof, *is used by* an architect, and *is applied to* the outline conceptual design. This leads to a precise search result. Figure 7-40 demonstrates the end-user interface of the SWIMS Document Search Engine. Project partners can input their query requirements through it.

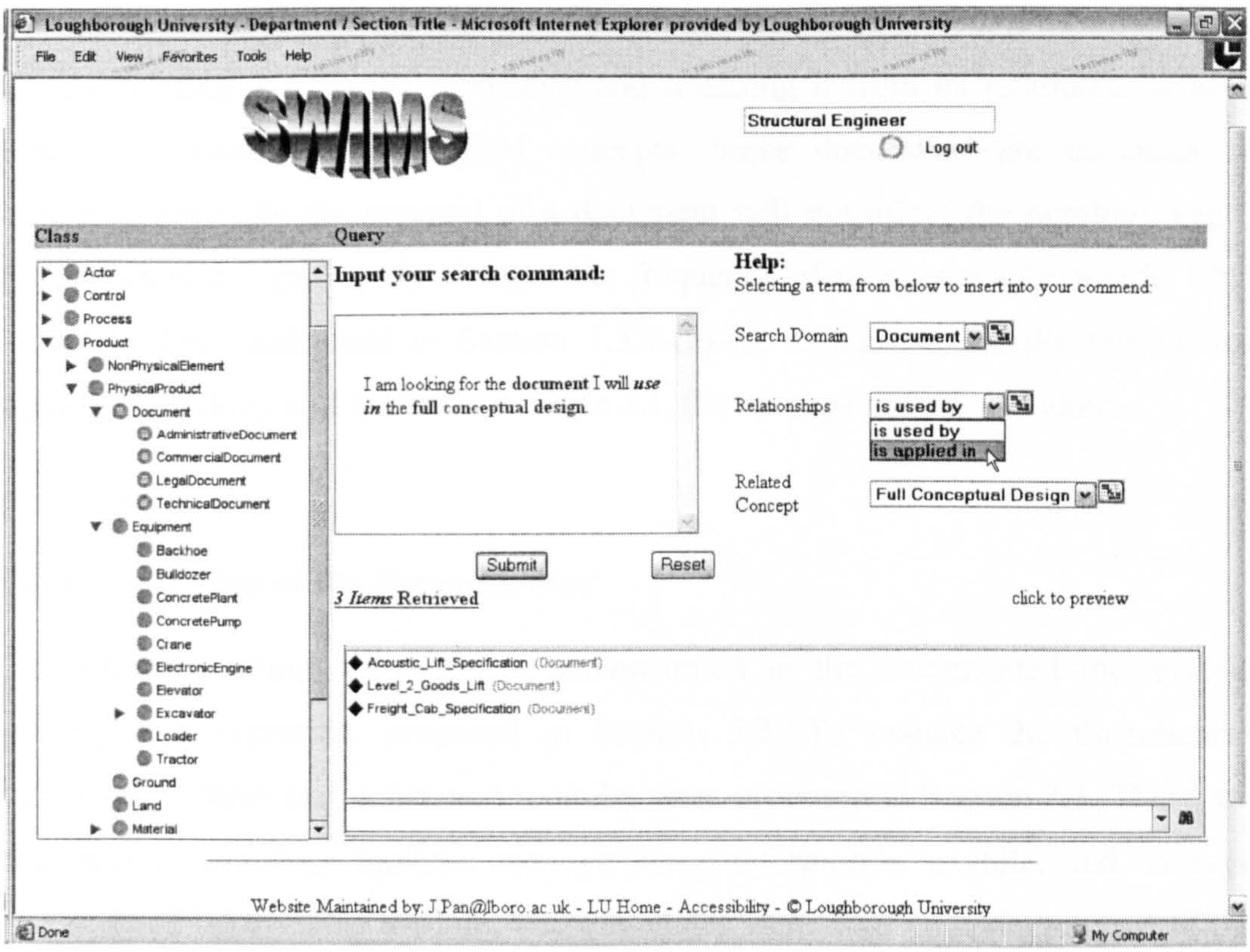


Figure 7-40: End-user Interface Template of SWIMS Document Search Engine

(3) Report Generation

Building construction is a dynamic process. It is often time consuming to manually rewrite related documents after information is updated. In SWIMS, documents, especially textual documents are recorded in XML/RDF transcript. Since computers can process every piece of XML tag and RDF statement, the generation of project reports can be undertaken by computers. Project partners can set up their blueprint and layout for a report, which specifies the outline of the report, the documents that this report refers to, and the concepts, properties and value types to be extracted from the source documents. The Semantic Web tools will then extract related information from the source documents, parse it and fill it into the report's body text according to the blueprint. This task integrates a set of Semantic Web tools, such as the information extractor software, the RDF parser, the ontology editor and the report generator.

(4) Remove Document

In SWIMS, most outdated documents are annotated by an old version number. Only those considered obsolete are removed from the system. The removal of a document means deleting it from the repository and releasing it from its relationships with other documents and ontological concepts. Since documents are instances of concepts, normally the removal of a document will not affect the ontology itself, except when it significantly changes the frequency of a concept's keywords. This issue has been addressed in Section 7.3.3-(2)-(i). If this occurs, the request for updating ontology will feedback to Node A1, the *manage ontology* module.

7.4 SUMMARY

7.4.1 Conclusion of the Demonstrator

The implementation of SWIMS is demonstrated in the documented information management scenarios, proposed in Section 5.3. To manage the documented information, three major function modules were proposed in Section 7.3. These are the *manage ontology* module, *manage actor information* module, and *manage documented information* module. These modules were used in managing part of the Royal Festival Hall Extension Building documents. In this case, the information repositories involved 68 project documents (19 of these were adopted in the demos)

from 19 organisations and 29 people. Both project information and partners' information were annotated with their properties by ontological concepts in SWIMS. Two annotation models were presented, the metafile for documents and the user profile for project partners. Information distribution and retrieval was based on these annotations and the associated/relevant ontologies. Project partners' ontologies and project's ontology could be mapped, merged and updated. The interaction between ontologies penetrated the boundaries of various information sources and applications, making information much more accessible than hitherto.

7.4.2 The Main Functions and Benefits of Adopting SWIMS

(1) Functions

(i) Content-based Information Management

- Construction information is well defined by the SWIMS ontology. The management and use of ontologies with deductive rules are being undertaken by computers, which support Web Services and optimize responses to meet query criteria.
- The system can process information and RFIs based on the meaning (i.e. semantics) rather than their textual expressions. Synonyms and technology standards are convertible within SWIMS. These lead to both attribute-based and knowledge-based document searches.
- The content of documents is accessible without the need for humans to read the documents.

(ii) Context-aware Information Services

- The system can consider the context of project partners (i.e. role, time, location, preferences) and provide the most appropriate information to them.
- Updating alerts can be sent to relevant partners based on the content of information and context of people.

(iii) Interoperable Information Management

- Project partners can customise their information management activities as well as keep the consistency of information throughout repositories.

- Project partners (and their agents) can share information via merging their speciality ontologies. The sharing is based on understanding the meaning of information in partners' information repositories, rather than the simple delivery of documents to partners.
- A reliable set of links between documents can be built using SWIMS. The links reflect the relationships between documented information and will not be affected by how the project partners categorised the files.

(2) Benefits

The functions of SWIMS would bring the following benefits to the construction sector:

- The use of SWIMS search engine considerably speeds up the retrieval and management of project documents.
- The ontology merger, terminology conversion, and the use of user profiles can reduce the risk of information overload, loss and misunderstanding.
- The report generation function reduces the time to generate reports about real-time project information.
- The semantic-based document management function enables the project information to be consistent throughout diverse information repositories.

In summary, the content-based information management, context-aware information services, and the interoperable information sources in SWIMS enable the co-operation between distributed project partners, which overcome the differences of time zones, terminologies and technical standards, so as to ensure the efficiency and precision of information management.

7.4.3 IT Tools Used

(1) Software

In this research, the implementation of SWIMS is demonstrated in the Protégé ontology development environment. Although, it is not intended to be an end-user software, it explicitly presents the technique and details of how to process project information and sort out the users' commands from an ontology engineer's

perspective. An interface of SWIMS document search engine is displayed in Figure 8-40. Referring to it, more sophisticated interfaces could be developed so as to hide the complicated background techniques and provide a more friendly information management environment to project partners.

(2) Hardware

The SWIMS demonstrator was programmed on a Toshiba A10 laptop with Microsoft Windows XP operating system. The final SWIMS application will be deployed on a group of computers which are connected through a network. Most electronic data receivers, such as desktop computers, laptops, PDAs and mobile phones, can be the terminal equipment for end-user access to SWIMS applications. Higher standard hardware may be required for the system server depending on the scale of ontology libraries and information repositories.

CHAPTER 8 - SYSTEM EVALUATION

8.1 INTRODUCTION

In the construction industry, project information is often dynamic, fragmented and heterogeneous. The Semantic Web has great potential to significantly improve the efficiency of construction project information management. To investigate how construction project information management could be developed within a Semantic Web environment, a Semantic Web-based Information Management System (SWIMS) was proposed for construction projects in previous chapters. The system framework was established, relevant techniques were specified, primary ontologies were developed and the implementation of SWIMS was conducted in the documented information management scenarios. As an integral part of the model development, the system evaluation process is described in this chapter.

Firstly, the objectives of the evaluation are outlined, and then the criteria applied to evaluate each objective are specified. Secondly, the methodology adopted for the evaluation is explained, including the selection of evaluation techniques, choice of evaluators, programming demos, and the design of the evaluation questionnaire. The evaluation of the system using demos is then presented with details of the evaluation process and an analysis of the responses. The analysis covers all the major aspects of the system, including the respondents' background, the features of the system, the feasibility of the system, the suitable application areas, and the enablers and barriers to the implementation of SWIMS. Finally, the result of the evaluation is discussed, as is the effectiveness of the adopted evaluation method.

8.2 EVALUATION OBJECTIVES

The aim of the evaluation is to determine how well SWIMS has potential to support construction information management, in terms of providing precise information to end-user applications by the middleware components. The study of information systems can be regarded as a multi-disciplinary endeavour in general (Land, 1993; Avison and Fitzgerald, 1991). The interdisciplinary research presented in this thesis addressed problems in construction engineering using Web technologies. Different from traditional social science or experimental projects, the criteria used in

evaluating SWIMS have two perspectives: one from the practice in construction information management and the other from information technology (IT); both of these are reflected in the system evaluation objectives and criteria as follows:

Objective One: Assess the Quality of the SWIMS Framework.

Existing Semantic Web-related research in the construction sector are fragmented. In this research, SWIMS is proposed to integrate these fragmented Semantic Web software, construction information repositories, construction information ontologies, and end-user applications. The criteria to assess the quality of the SWIMS framework include:

- **Feasibility:** Can construction information be managed in a Semantic Web environment, and can SWIMS successfully support it?
- **Compatibility:** Is this system framework compatible to existing information management tools?
- **Sustainability** and **extensibility:** Are the metafile and ontology reusable within the SWIMS framework? Is the SWIMS ontology extensible? Can it be merged with other ontologies?
- **Efficiency:** Does SWIMS significantly reduce the information-processing time for project partners?
- **Simplicity:** Is SWIMS easy to use for project partners?

Objective Two: Assess the Performance of SWIMS Deployment

The deployment of SWIMS was conducted in a documented information management scenario. Lightweight ontologies, annotation models and deductive rules about project document are developed. They were applied to manage a small amount of documents from the Royal Festival Hall Extension Building project within the SWIMS framework. Four demos were edited to present the operation of SWIMS, enabling the assessment of the deployment's performance. The criteria to evaluate SWIMS for this objective are:

- **Rationality:** Does the SWIMS ontology and metafile best reflect the characteristics of construction information in the proposed scenario?

- **Improvement**: What features can be integrated into the system to enrich SWIMS?

Objective Three: Assess the prospective industry applications of SWIMS

Identifying prospective industry applications and the end-user's expectations are valuable for carrying out further research. The criteria to evaluate SWIMS for this objective are:

- **Suitable application domains**: Can SWIMS actually benefit users compared with traditional information management systems? Which areas in the construction industry would get most benefits from the use of SWIMS?
- **Advantages** and **limitations** of SWIMS: From a user's viewpoint, what are the advantages of SWIMS over traditional information management systems and what limitations still exist?
- **Enablers** and **barriers** for adopting SWIMS.

8.3 METHODOLOGY

To achieve the above system evaluation objectives, the following steps have been taken:

8.3.1 Select Evaluation Techniques

Miles et al (2000) suggested two evaluation methods that are popular for knowledge-based systems:

Case studies: Using this method, "a trial of the system is provided to industry users and leaves them to make use of it over a prolonged time period (e.g. a number of weeks). Evaluators use a diary to record their practices including information on any difficulties that occurred and any features that are felt to be lacking" (Miles et al, 2000). For example, the On-To-Knowledge project used three cases to evaluate its methodology and toolkit (On-To-Knowledge, 2002). The advantage of the case study method is that the evaluators are given an opportunity and enough time to "get used to the system's functionality and form an opinion on whether the stated benefits are actually achieved". The limitation of the case study method lies in the difficulties involved with "carrying out a trial in a working environment", because

most evaluated systems are immature. The difficulties are "magnified when the trial system must be integrated with existing systems" (Ren, 2002). Also, it is difficult to find organisations or groups of people prepared to take part in the experiment.

Focus group: Using this method, "an evaluation workshop is held in a single location with all the evaluators participating simultaneously. The evaluators are then guided through a usage scenario with the use of appropriate notes. The workshop is followed by the distribution and completion of a questionnaire by each evaluator. The questionnaires will be used to collect evaluators' opinion on the system" (Miles et al, 2000). The advantages of the focus group method are: evaluators can interact with the system designer; it guarantees the rate of feedback; the cost and risk of evaluation practice are lower than case studies. However, this method requires a relatively large number of evaluators to be available for a short period of time. Although there is face-to-face guidance, evaluators may not understand and become familiar with the system in a relatively short time.

Considering that the Semantic Web is a new technology, it takes time for evaluators to understand the technology and evaluate the system. A Web-based demonstration method, which includes the advantages of the above two methods, was used to evaluate SWIMS. It includes Webpages, which introduce the SWIMS framework, and videos, which present the deployment of SWIMS in a sample project. The Web link of the demonstration was sent out to selected evaluators through e-mails. Technical support (via e-mail and telephone) was available to evaluators while an on-line questionnaire was used to collect feedback.

The advantages of the Web-based evaluation method are as follows:

(1) The Web-based demonstration allows evaluators to go through the demonstration Webpages step by step and freely repeat the demos. They can evaluate it at their convenience and thus have enough time to understand and familiarise themselves with the system, to ensure reliable results.

(2) Evaluators do not have to be in the same place, which is more flexible than the focus group method. This can reduce the evaluation cost significantly and can involve more evaluators.

The limitation of the Web-based evaluation is the difficulty to ensure a high feedback rate. Return rates for e-mail invitations to a Web-based survey are generally less than 10%, depending on the specifics of the study (Hooper, 2003). A study on how to improve the response rate was carried out and some solutions were identified, such as using personalised salutations and ensuring that e-mails were sent by someone already well-known and respected in the field (Joinson and Reips, 2005).

8.3.2 Choice of Evaluators

Considering the specific characteristics of this research, two criteria were used in choosing evaluators:

(1) Rich experience in IT and construction information management

Since the SWIMS deployment examples were presented in Protégé, which is not an end-user application, it will take considerably more time for people without the above background to understand and evaluate the system.

(2) Wide representation of stakeholders

Ideally, evaluators should cover all project partners, ranging from clients to site managers, and involve all stakeholders of an organisation. This ensures the evaluation results are representative viewpoints from different levels and type of users.

The evaluation panel was selected following the above two criteria. Twenty people from industry and academia were invited to the evaluation. They had various industry/academic backgrounds and experience of information management in construction. The academic experts included professionals in knowledge management, design-construction integration, mobile communication, construction site management, and structural engineering. The industry experts represented various project partners, such as: contractors, consultants, engineers and architects; and held various positions in their organisation, such as: project director, collaboration consultant, research engineer and senior engineer. Meanwhile, in order to evaluate the pure IT technologies used in this research, an IT developer was invited.

8.3.3 Design of Evaluation Demos

Since the study on the Semantic Web, especially for its application in construction, is quite new, the pre-requisite at this stage is to gain an understanding of whether the Semantic Web technology is appropriate and applicable to the construction industry and what the end-users' expectations for the Semantic Web-based applications are, rather than have industry users try out a specific application. Therefore, the demonstration developed in this research focused on presenting the generic capabilities of the Semantic Web. To present the advantages of SWIMS over traditional information management systems, it is necessary to instantiate the use of SWIMS. The instances should be generic, typical, simple, explicit and demonstrative. According to these criteria, Webpages (static technique) are used to present the SWIMS framework, and movies (dynamic technique) are used to present SWIMS deployment. The demonstration contains four parts:

(1) Introduction: Background knowledge and scenarios used in following demos.

(2) Demos: Four Demos instantiating the operation of SWIMS within a documented information management scenario. For convenience, controls were available to play, pause, or stop the movie.

(3) Discussion: The benefits of adopting SWIMS in construction project information management and the readiness of the industry.

(4) Evaluation Questionnaire.

Meanwhile, the confidential issue and the reliability of responses were considered because the demonstration is published on the Internet. The demonstration Webpages were protected by passwords. Only invited experts were granted a username and password to access the Website. This prevents unqualified/illegal respondents.

8.3.4 Design of Evaluation Questionnaire

The evaluation questionnaire consists of 50 questions categorised into 4 sections:

- The background of the respondents in terms of their IT experience and practical experience in the construction industry, the role(s) they have held, and the area(s) of experience;

- The feasibility, sustainability, suitability, efficiency, simplicity of the SWIMS framework and major components;
- The rationality, improvement of the SWIMS deployment examples;
- The suitable application areas of the system, advantage and limitation of the SWIMS model, and the enablers and barriers for adopting SWIMS to support construction project information management.

The full version of the questionnaire is attached as Appendix E. The questionnaire used a mixture of multiple-choice and open-ended questions. For many multiple-choice questions, a five-point scale was used. The scale was similar to the *Likert scale* (Likert, 1932) but used terms rather than agreement and disagreement. The rating levels were presented to the evaluators as 1. Poor, 2. Fair, 3. Satisfactory, 4. Good and 5. Excellent. The five-point scale was adopted as it was deemed to provide a sufficient range of responses without being overly complex. Also, the use of a scale categorises evaluators' attitudes into a quantified range, making the results easier to be compared and analysed.

For open-ended questions, respondents were allowed to make specific comments on the SWIMS framework, components, deployment examples and the prospect of industry applications.

8.4 EVALUATION OF SWIMS FRAMEWORK AND DEPLOYMENT EXAMPLES

8.4.1 Evaluation Process

The evaluation was carried out adopting the Web-based demonstration method specified in Section 8.3. Invitation e-mails were sent out to 20 experts, including an explicit introduction to the research and links to the demonstration Webpages with a suggested timeframe for completing the evaluation. The invitation also used personalised salutations and mentioned people who recommended the evaluators to the researcher, so as to enhance the response rate of questionnaires. After each questionnaire was submitted, the system would send an automatic notice e-mail to both the author and evaluators. Feedback was obtained via these questionnaires. Every two weeks, a follow up e-mail was sent out to the irresponsive evaluators

urging them to return the questionnaire. The whole evaluation process took 2 months.

8.4.2 Analysis of Results

Using the feedback from the questionnaire, an analysis on various aspects of the SWIMS model was conducted. Opinions of six academic experts and five industry experts were obtained. The findings presented here are based on an effective 55% return rate (11 out of 20). During the evaluation process, an intensive discussion with an industry evaluator took place via e-mail, which was of great help for future improvement.

The details of the evaluation results are presented below:

(1) Evaluators' Background

(i) Construction industry background

It can be seen from Table 8-1 that respondents covered many domains and held a wide range of roles in academia and industry.

Table 8-1: Evaluators' Industry Background

Specialty		Number of Invitee	**Number of Respondent**
Academic	Mobile communication in construction	1	**1**
	Knowledge management	2	**1**
	Construction site management	1	**1**
	Design-construction integration	2	**1**
	Computer-Aided Engineering	3	**1**
	Structural engineering	1	**1**
Industry	Architect	2	**1**
	Engineer	1	**1**
	Construction	2	**1**
	Consultant	4	**1**
	Other (Web Engineering)	1	**1**
Total		**20**	**11**

Table 8-2 shows that evaluators have a good work experience in the construction industry. Seventy three percent of the evaluators had over three years of related experience. The only one who had less than 1 year work experience was an evaluator from the Web Engineering discipline. Amongst those from the construction industry, 80% of them had more than three years work experience.

Table 8-3 shows the positions that evaluators hold in their organisations. Since at the current stage, evaluations from diverse project partners' viewpoints to the rationality of proposed concept models were considered to be of the most importance, evaluators were selected from different project projects/organisations rather than all stakeholders of a single project/organisation. Respecting the positions the evaluators hold, it is reasonable to assume that the opinions of this evaluation were representative for different levels of stakeholders in organisations.

Table 8-2: Evaluators' Related Work Experience

Years of work (years)	Less than 1	1-3	4-6	7-10	Over 10
Number of Respondent	1	2	3	1	4

Table 8-3: Evaluators' Position in Their Organisations (excluding the IT expert)

Job Title	Researcher	Lecturer	Consultant	Engineer	Project director
Number of Respondent	3	3	1	2	1

(ii) Information management tools background

Information management tool experience is very helpful for evaluators to identify the advantages and limitations of SWIMS. A question was designed to collect evaluators' experiences of using other information management tools. As shown in Table 8-4, 73% of the respondents were familiar with manually indexing document systems, 91% of them were familiar with electronic information management systems, and 64% of them were familiar with Extranets.

Table 8-4: Evaluator's IT Background

Have Experience in Using Information Management Tools	Number of Respondent	**Percentage**
Traditional information management systems (e.g. manually indexing project documents.)	8	**73%**
Electronic information management systems (e.g. enterprise wide information management system)	10	**91%**
Extranets (e.g. Asite, BIW and 4Projects)	5	**64%**

Based on the above information about the respondents' roles played, positions held and experiences in both the construction industry and IT, it was assumed that the evaluation panel were sufficiently qualified to provide a fair assessment on the system.

(2) Responses to Questions about SWIMS

Many open-ended questions and five-point scale questions were used to collecting evaluators' opinions on the system model, including the framework, major components, and the use of the system model.

(i) Responses to questions about SWIMS framework and components

This section contains analysis of questions about various aspects of the overall SWIMS framework and components, such as their feasibility, compatibility, sustainability, extensibility, efficiency and simplicity.

Compared to the other types of existing information management tools available, 73% of respondents thought the Semantic Web-based system had advantages over the tools they had previously used. Tables 8-5 to 8-7 show the respondents' opinions on the SWIMS framework and components in detail. The questions and responses were categorised in accordance with the evaluation criteria.

The responses to Questions 1-3 (Table 8-5) confirmed the feasibility of the SWIMS framework to construction information management activities. The evaluators agreed that:

- SWIMS *framework* can adequately support construction information management activities;
- The *concept* model can well represent construction project information;
- The concept hierarchy and *ontologies* represent the relationships between construction information well.

Respondents also expressed doubt as to whether SWIMS is compatible with existing information management tools, as shown in Question 4 (Table 8-5). This issue has been considered since the beginning of the framework development (addressed in Sections 5.2.1-(1) and 5.2.3). However, it seems that end-users' doubts were not eased by the theoretical model. It is normal that users will not be convinced unless

they really see it happen. Technically, much more work needs to be done to ensure the compatibility of SWIMS in the future.

Table 8-5: Responses to Feasibility and Compatibility of the SWIMS

Opinion Scale / Number of Respondent / Questions	1	2	3	4	5	Mean
(1) How well does the *framework* support construction information management activities?			2	7	2	**4.00**
(2) How well do you think construction project information is represented in the *concept* model?			3	6	2	**3.91**
(3) How well are the relationships between construction information represented by the concept hierarchy and *ontologies*?			3	7	1	**3.82**
(4) How compatible is the *framework* with existing construction information management systems?		2	3	6		**3.36**

* Scale: 1. Poor, 2. Fair, 3. Satisfactory, 4. Good, to 5. Excellent.

Data obtained from Questions 5-8 (Table 8-6) indicates that most respondents agreed that SWIMS is sustainable.

Table 8-6: Responses to the Sustainability of SWIMS and Efficiency of Use

Opinion Scale / Number of Respondent / Questions	1	2	3	4	5	Mean
(5) How well do you think the *concept* and *ontologies* are reusable?			5	3	3	**3.82**
(6) How well do you think the flexible and extensible architecture of the *concept* hierarchy and *ontologies* enable information to be shared between diverse communities?		1	1	6	3	**4.00**
(7) How useful are *metafiles* in saving time to access the information contained in documents?			3	7	1	**3.82**
(8) How useful are *concepts* and the *ontology* in avoiding mistakes during the information exchange process?		1	4	4	2	**3.64**

* Scale: 1. Poor, 2. Fair, 3. Satisfactory, 4. Good, to 5. Excellent.

A fairly high score was given to the reusability and extensibility of the *concept* and *ontologies* (Questions 5 and 6). Evaluators also thought the use of SWIMS can enhance the efficiency of information management activities. Most of them agreed on the techniques used "enable information to be well shared between diverse communities" (Question 6) and "the *metafiles* are useful in saving time to access the information contained in documents" (Question 7). However, the response of Question 8 only got a fair neutral score. This indicates that the respondents did not think the use of the Semantic Web could significantly (just satisfactorily) avoid mistakes during the information exchange process. This can be explained by the fact that information misunderstanding is not the only factor which leads to mistakes. Not all of factors can be addressed and resolved by the use of the Semantic Web, like social or emotional factors.

The evaluators were also asked to indicate the ease of use with SWIMS. The operation of SWIMS was broken down into a series of actions, as shown in Questions 9-16 (Table 8-7). The percentages in Table 8-7 express the number of evaluators, whose responses fall into the range of 'easy' to 'very easy' (4 - 5 in the five-point scale), relative to the total number of respondents.

Half the respondents thought it was easy to specify *concepts* which the information is accompanied with. Fifty six percent of respondents thought relating a *concept* to existing concepts was easy. Sixty percent of respondents thought it was easy to define a *concept* and use a semantic *search* service to make a query. Eighty percent of respondents thought it was easy to edit *ontology* for concepts and add *individuals* to concepts. Ninety percent of respondents thought it was easy to specify *metafiles* to concepts. Using OWL reasoners to make a *reasoning* seems a bit more difficult than the other actions. However, the data range for this question (Question 15) was distributed. One respondent thought it was difficult while two thought it was very easy. This result may be partly due to the variety of expertise and experience of respondents. The finding indicated that the use of the OWL reasoner would be a key task when training project team members to use the system in the future. Although it might not be easy to use the reasoning function, most respondents admitted that the inferring ability is a useful feature, as indicated in Questions 19-21 (Table 8-8). This issue will be discussed later.

Table 8-7: Responses to the Simplicity of Using SWIMS

Opinion Scale / Number of Respondent / Questions	1	2	3	4	5	Percentage of Responses with Scale 4-5**
(9) Specifying *concepts* which the information is accompanied with;			5	5		**50%**
(10) Defining a *concept*, e.g. documents, structural elements and actors;			4	4	2	**60%**
(11) Relating a new *concept* to existing concepts;		1	3	4	1	**56%**
(12) Specifying *metafiles* to concepts, e.g. Actors have attributes like name, e-mail, role, and interests;		1		6	3	**90%**
(13) Editing *ontology* for concepts, e.g. Actors are disjoint with Documents;		1	1	6	2	**80%**
(14) Adding *individuals* to concepts, e.g. Mr. Smith is an individual of Actors;			2	6	2	**80%**
(15) Using OWL reasoners to make a *reasoning*, e.g. consistency checking and inferred ontology computing;		1	5	2	2	**40%**
(16) Using semantic *search* service to make a query, e.g. selecting searched range and concepts.		1	3	5	1	**60%**

* Scale: 1.Very Difficult, 2. Difficult, 3. Fair, 4. Easy, 5. Every Easy.

The data obtained from the above questions gave a fairly optimistic result. It seems that most respondents can easily handle the basic operations on SWIMS. This indicates SWIMS are not difficult to understand and operate to professionals with backgrounds in IT and the construction industry. Considering at the current stage, the software used to introduce SWIMS is not an end-user tool and is for knowledge/ontology engineers only, it can be expected that SWIMS tools/applications with a user-friendly interface will be easy to the project partners who have basic computer skills.

(ii) Responses to questions about deployment examples

The deployment of SWIMS was instantiated in the documented information management scenario. Three examples were presented in the demos:

- Automatic document categorisation example;
- Terminology conversion example;
- Content-based (attribute-based) document search example.

Respondents were asked to answer the questions about the rationality of the deployment examples and ways to improve the use of SWIMS. As shown in Table 8-8, relatively high scores were given to most of the classification and inference-related features (Questions 19-21). The benefit of SWIMS in information search was also affirmed (Questions 22-23).

Table 8-8: Responses to the Rationality of the SWIMS Deployment Example

Opinion Scale / Number of Respondent / Questions	1	2	3	4	5	Mean
(17) How well is the content of construction project documents and project partners' information captured in *metafiles* and *user profiles*?	1	1	3	4	2	**3.45**
(18) How well are the *classification* and *inference* functions applied to construction information?			5	6		**3.55**
(19) How useful are the *classification* and *inference* functions in keeping information consistent between the project partners?			2	7	2	**4.00**
(20) How useful are the *classification* and *inference* functions in distributing timely information?		1	2	6	2	**3.82**
(21) In general, how useful are *classification* and *inference* functions in improving the efficiency of construction information management?			2	8	1	**3.91**
(22) How well does the semantic *search* help in understanding the meaning of a RFI?			1	8	2	**4.09**
(23) How well does the semantic *search* help project partners get appropriate and timely information?		1	2	7	1	**3.73**

* Scale: 1. Poor, 2. Fair, 3. Satisfactory, 4. Good, to 5. Excellent.

Respondents commonly agreed that the *classification* and *inference* functions are useful in improving the efficiency of construction information management (Question 21) and keeping information consistent between the project partners

(Question 19). The semantic *search* mechanism is helpful in understanding the meaning of a RFI (Requests For Information) (Question 22). However, respondents gave relative low scores to the annotation-related features (Question 17-18). They were not very confident on capturing the content of construction project documents and project partners' information into *metafiles* and *user profiles* (Question 17). Respondent also found it difficult to configure the system so as to enable the construction information inferable (Question 15 and 18). Their worries are understandable. Firstly, the construction information is characterised by fragmentation and poor structure. Although some relationships of concepts/instances have been identified in this research, such as the interdependencies of documents, exploring all possible logical relations between construction information is an enormous task; one which is beyond the scope and resources of this work. Secondly, information reasoning is not as straight forward as information searching. It may take a long time for people to get familiar with it.

In short, data from the responses showed the rationality of the SWIMS deployment examples was good. The result also indicated that well established SWIMS can improve construction information management. However, much work should be done to enable information being annotated within SWIMS and to train people to use SWIMS, especially the inference-related operations.

After reviewing the demonstrated deployment examples, evaluators were asked to propose suggestions to improve the preliminary version of SWIMS (used in the deployment examples), via an open-ended question. Five respondents affirmed the rationality of the demonstrator and explicitly indicated that the developed construction information concepts and ontology would be better if more concepts/ontologies were added to it. Respondents said:

> "*The ontology is quite small and would doubtless need extending to satisfy the complex heterogeneous construction environment.*" (Evaluator B1)

> "*I think the developed concepts/ontology is good enough for demonstration purposes. Any real life deployment will have specific requirements.* " (Evaluator A1)

> "*As a demonstrator the produced ontology is sufficient to demonstrate the proof of concept.*" (Evaluator A6)

"It did what it set out to do" (Evaluator B3)

Two other respondents specified the concepts/ontology they would like to see:

"Post project review reports, case studies, explicated knowledge files." (Evaluator A2)

"Progress of the work; health and safety plan; monitor cost and budget." (Evaluator A3)

It was indicated in the ontology-development methodology: *"There is no single correct ontology for any domain"* (Noy and McGuinness, 2001). This principle is coincident with Evaluator A1's opinion. Any specific deployment (domain) will have specific requirements and that will undoubtedly affect ontology. The most important issue at this stage is ensuring the extensibility of shared ontology and the compatibility with speciality ontologies within the SWIMS framework. These abilities are basic features of ontology and have been acknowledged. As Evaluator A6 said:

"The main strength of the developed ontology is that it is extendable."

Therefore it can be enriched referring to evaluators A2 and A3's suggestions. There is great scope for enriching SWIMS Ontology 1.0 and developing other speciality ontologies. Also, based on a mature system model, a case study method is appropriate for identifying more practical concepts and ontologies. *"It will be useful to apply demonstrator in a real life project"* (Evaluators A2 and A3).

From another perspective, Evaluator A6 proposed a suggestion for improving the system:

"The system is good but passive in its current stage. It does not prompt the user/ push any information as per his need basis. This could be an added feature in later systems?"

This comment was very crucial. The Semantic Web could provide the context-aware Web services to project team members. It was designed in the scenarios but not directly displayed in the demonstrator for two reasons: Firstly, this service requires developing an end-user application. However, evaluation of proposed concept models was considered to be of more importance than user interface

application at the current system development stage. Secondly, what has been done and shown in this research was the basis of context-aware applications. Since the information could be filtered according to user's preferences and other semantic constraints, it could be followed by a context-aware service. This would be a good topic for further researches.

(iii) Responses to questions about industry applications

This section contained questions about the advantage and limitation of the SWIMS model, and the enablers and barriers for adopting SWIMS to support construction project information management.

The evaluators were asked to select three areas which will get the most benefits from SWIMS in their opinion. Table 8-9 shows the suggested areas and their rank.

Table 8-9: Responses to the Suitability of SWIMS

Questions	Number of Respondent	Rank
Information Management and Exchange		
Information collection to facilitate decision making in the pre-design stage, such as feasibility study;	3	**11**
Information sharing between design team members;	8	**2**
Information distribution and collection on the construction site (such as the retrieval of up-to-date drawings);	3	**11**
The collection of information for project inspection and retrieval of defect information;	4	**9**
Information retrieval during building maintenance;	7	**4**
Collaborative Communications		
The communication between project team members;	8	**2**
Procurement		
E-commerce for procurement of goods and services for the project;	6	**5**
Health and Safety Management		
Distributing health and safety documents to related project team members;	5	**7**
Knowledge Management		
Post-project evaluation;	4	**9**
Knowledge management on a personal basis;	5	**7**
Project knowledge management;	9	**1**
Organisational knowledge management;	6	**5**
Others areas specified by evaluators		
• "*The mind maps will help the individual find their way through the labyrinth*" (Evaluator B3) • "*Reduce cost*" (Evaluator B4)		

Because SWIMS has broad potential application areas, 91% of respondents selected more than 3 areas. As Evaluator A6 asserted:

"The listed uses are all useful. It is difficult to predict which would be the most useful without actual field study or some feasibility analysis."

The rank of each suggested application areas indicates that the use of SWIMS would benefit the project knowledge management most, and then the information sharing and communication between project team members.

After assessing the system from various aspects, six respondents explicitly declared that they would consider adopting the system in their organisation, while four not (one respondent didn't answer the question). These declarations of attitude were followed by two open-ended questions. Respondents were asked to specify their opinions on the advantages and limitations of the system, and enablers and barriers of adopting the system.

- The advantages were specified as follows:

  "The inferring capability of the tool, i.e. about to automatically include the search results with synonyms of the items searched for." (Evaluator A2)

  "The best feature is a shared ontology and semantic standards. It will help project management team to share information (e.g. drawings, specification) across devices, platforms and applications." Also, *"it will be helpful for the construction project manager and consultant to manage the construction project."* (Evaluator A3)

  "The search options and the way in which several filters can be used to acquire isolated information" are advantages (Evaluator A6).

- The following items were identified as the enablers for adopting the system:

  "Current trends towards real-time enterprise, knowledge management, intelligent systems" will enable more companies accept the system (Evaluator A1).

  The *"need for better information management"* will enable the system being adopted (Evaluator B5).

The "*support from the senior management for essential resources.*" will be an enabler (Evaluator A2).

The support from the client will be an enabler. "*The client needs to make sure their consultant (e.g. structural engineer, architect and mechanical engineer) use the system before being implemented to the project management system*" (Evaluator A3).

Other enablers include: "*1. Faster and more accurate searching, 2. Better KM system, 3. Information being assimilated even though it might exist on several different databases*" (Evaluator A6).

The features of "*easy to learn, simply to use, convenient to upgrade, compatible to existing databases*" are helpful to make the system accepted by industry (Evaluator B4).

- The limitations of the system and the barriers to adopting the system focused on three points:

<u>The extra effort in developing and maintaining the ontology</u> was a focus of the discussion:

"*The effort required to build and maintain the ontology. The effort required to implement systems and processes to ensure that documents are marked-up and maintained either automatically or manually.*" (Evaluator B1)

"*Too much addition effort is required to create and maintain an ontology. This must be weighed against the benefits that the system brings later in the process for searching and retrieving information. It would be more attractive if less effort was required to create concepts, properties, etc.*" "*Overhead of creating and maintaining an ontology.*" (Evaluator B3)

"*My main concern is the amount of time required for data preparation*" (Evaluator A6).

<u>The extra cost for adopting the system</u> was another concern:

"*Involve a learning curve, costs of implementing such as system (compared to indexing based current approaches).*" (Evaluator A1)

"*Cost for IT resources (including staff time).*" (Evaluator A2)

The readiness of the current construction industry for adopting the system was also addressed:

> Evaluator A3 indicated this issue in two considerations: "*infostructure (e.g. how to standardised the information between all parties), and infrastructure (e.g. hardware, intranet system, server, etc.)*". "*The training of the staff to use the system also presents the barriers*"
>
> Evaluator B2 mentioned "*A lack of definition and classifications-links between different terminologies*" currently will be a barrier. But he also indicated that the "*existing attempts to develop industry classifications for all aspects of the construction industry (i.e. uniclass)*" should be considered. The reuse of existing classification systems/taxonomies has been argued during the ontology development process. It can refer to Sections 3.6.3 and 3.6.5.
>
> Evaluator A6 said my "*other concern would be how do you get the industry to accept a common terminology?*"

The evaluators' considerations were true. Nineteen real project documents from the Royal Festival Hall Building project were used in the demonstration. The amount was limited because at the current stage, the documents were annotated manually. It would take considerably more time to annotate large volumes of documents because existing extractor software can not completely meet the needs in this research (as reviewed in Section 5.2.3). Therefore, only 19 documents were used in the demonstrator (68 documents were used in the research). Although it was adequate to present the deployment of SWIMS, it would be a barrier to full scale deployment.

> Evaluator B3 thought his "*company has a corporate system that it uses and it would not be easy to change it.*"

It is true that making any change in an existing organisation is not easy, unless the benefit from the change is significant. The industry cannot get direct benefits from SWIMS now because the system is still immature. It can hardly provide industry users with satisfactory end-user interface applications and it did not cover all information repositories in the first version SWIMS. So the advantages of the Semantic Web techniques can not be fully presented. The aim of this research is to investigate how construction project information can be managed within a Semantic

Web environment. As long as the rationality of Semantic Web-based information management system was proved, further efforts could be put on developing more realistic SWIMS toolset/applications which could provide more satisfactory solutions to real problems. It should "*show the friendly interface to user who is not a professional and easy to learn*", as expected by Evaluator B4.

(3) Responses to Questions about the Effectiveness of the Evaluation Approach

A new approach was used in this research to evaluate the quality and performance of SWIMS. Different from other methods, it was a Web-based multi-media approach. The reasons why this method was adopted were specified in Section 8.3.1. The anticipated advantages were confirmed by respondents in their feedback.

> Evaluator A1 said "*It is very well-presented. I found graphical presentation and software video and commentary very useful*".
>
> Evaluator B1 thought "*The visualisation*" of ontology is good. "*This allows non-expert users to understand the ontology*".
>
> Evaluator B2 and B3 also though "*The talk through was good*". The demos were "*overall extremely professional and well presented.*"

Three suggestions were given by four respondents:

> "*It might have been useful to provide some definitions of terminology used.*" (Evaluator B2)
>
> The proposed technical issues to improve the evaluation materials include using "*lesser screen resolution*" (Evaluator A5) and "*might be good not to use music background*" (Evaluator B2)

The return rate of the questionnaires was 55%, much higher than the average return rate of on-line surveys (10%). Generally speaking, the Web-based multi-media evaluation method was effective.

8.5 DISCUSSION AND SUMMARY

SWIMS (the Semantic Web-based Information Management System for construction projects) was evaluated to accomplish the following two objectives:

- To assess the quality of the system framework;

- To assess the performance of the system deployment.
- To Assess the prospective industry applications of SWIMS.

Considering the techniques used in the system are unfamiliar to most project participants, a Web-based multi-media evaluation method was adopted. The invited 20 evaluators had good experience in both construction information management and the use of IT. Fifty five percent of evaluators went through the demonstration and completed the questionnaire. They came from both academia and industry, representing the diverse parties of construction information management, like experts in the design-construction integration, knowledge management, construction site management, construction project director, architecture, structural engineer, and collaboration consultant.

The overall response falls into the range of satisfactory to good (3-4 in the five-point scale). This may be due to the following three causes:

(1) A five-point scale was used in the questionnaire. Respondents may avoid using extreme response categories (*central tendency bias*). This issue is addressed in Section 4.2.3.

(2) The Semantic Web uses concepts, attributes, and relationships to represent information/knowledge. This changed the way of information representation which most people get used to. It is often difficult for people to acknowledge it at once.

(3) The Semantic Web is a practical technology, but is still in its infancy. Although many benefits to human's daily life brought by the Semantic Web have been enumerated, the Semantic Web itself is still too technical to be understood by the general public. Its component techniques perform tasks in a "black box" behind interface applications, which make it really difficult for end-users to envision what might be possible in the Semantic Web era before it advents. People hardly give it a high ranking unless they deem it a robust, steady, and friendly system to resolve their real-life problems.

Therefore, a satisfactory result is good enough to SWIMS at the current stage. Analysis of the responses proved that the SWIMS framework is feasible. It can support construction project information management. The compatibility of the

system is satisfactory, but it needs to be proven in real projects. The components of the framework are reusable and extensible. The system is most suitable for project knowledge management, communication and information sharing between project team members. The operations of using SWIMS are simple, but configuring and setting-up SWIMS ontology/metafiles are just of satisfactory ease. The efficiency of the system is satisfactory. The proposed deployment examples are rational. The consideration of reusing existing construction information classification systems can improve the performance of the system. The characters of the Semantic Web, such as the ability of reasoning, are useful in construction information management. However, too much effort is required to set up and maintain the system at the current stage. This would be weighed against the benefits that the system brings in the information management process. Also, much work needs to be done to build more speciality ontologies, use the system in real life projects, develop end-user friendly interface applications, and enable its compatibility with existing information management tools.

In short, the evaluations of SWIMS indicated that the quality of the proposed system and the rationality of deployment examples are satisfactory. This means the proposed Semantic Web-based system is beneficial to manage construction project information. The system can improve the efficiency of information management at a satisfactory level.

The enablers and barriers for the adoption of the system pointed out essential items that need more attention in the system development and implementation. These points will contribute to the recommendations for further development, which are addressed in the next chapter.

CHAPTER 9- CONCLUSIONS AND FURTHER DEVELOPMENTS

9.1 INTRODUCTION

This chapter first summarises the work that was carried out to achieve each of the research objectives. This includes a review of the background knowledge, model development process, the system model implementation, and evaluation. Conclusions are then drawn based on the research findings, and the limitations of the research are discussed. Recommendations are made for further work, which includes extensions to the prototype system, improvements to Semantic Web technology and tools, and preparations for future practical applications.

9.2 SUMMARY

Modern construction is an information-intensive process, characterised by its fragmented nature. The collaboration between diverse project partners requires an efficient and intelligent information management environment. The lack of a semantic-aware information management system is a barrier to achieving the effective collaboration between partners so as to increase the productivity of the construction industry. The Semantic Web provides a solution to the above problems. As stated in Chapter 1, the aim of this research was to investigate how construction project information can be managed within a Semantic Web environment. In order to achieve this goal, five objectives were defined. The specific tasks undertaken in this research with respect to the research objectives are:

Objective One: To Review Semantic Web Theories and Background Technologies

The review of Semantic Web theories, languages and techniques (as presented in Section 2.2) showed that Semantic Web is a relatively new Information Technology which aims to propel current Web-based information services into an automatic and interactive realm. The architecture of the Semantic Web integrates several existing Web techniques (such as Unicode, URI and XML(S)), several developing Web techniques (such as RDF(S), Ontology, Logic and Digital Signature), and several to-be developed techniques (such as the Proof and Trust). These techniques work together to enable the computer to "understand" and process information, which can

only be done by human beings on the current Web. Available tools to develop Semantic Web applications are also reviewed (as presented in Section 2.4). They are mainly of three types: RDF/XML Parsers, Ontology Editors and Ontology Reasoners. Based on a comparison of these tools, Protégé-OWL and RacerPro were adopted.

Objective Two: To Review Implementations of the Semantic Web Technologies in Other Industries in Order to Identify the Key Features, Enablers, Barriers, and Potential Benefits

In order to identify the key features, enablers, barriers, and potential benefits of applying the Semantic Web technology to the construction industry, related implementations in other industries were reviewed and compared (as presented in Section 2.3). The literature review revealed that the experiences of human resource management, medical and clinical information management, e-Commerce, publications and news management, and museum collection management demonstrated that the Semantic Web has great potential in content-based Knowledge Management and context-specific Web Services. These initial applications of the Semantic Web revealed the essential nature of the Semantic Web, how it works, and which technologies are being used or proposed for the Semantic Web's development and implementation. Referring to the experiences from industries, the key features, enablers, barriers and the benefits of implementing Semantic Web technologies were argued in Section 2.3.3. The efficiency of information management in those industries was improved by applying the Semantic Web technology, and vice versa, many Semantic Web tools and methodologies were developed in those industrial projects. The Semantic Web's development always kept a win-win relationship with its industrial applications. Because the Semantic Web is under active development now, most of the barriers to its implementation lie in the immaturity of the techniques. It is expected that, as well as in other industries, the Semantic Web application in the construction industry will bring benefits to both construction information management and the development of the Semantic Web. Based on the review of current Semantic Web-related research in the construction sector, six information-intensive construction management areas were identified as suitable for the use of the Semantic Web. These include construction knowledge management, collaborative design, on-line

procurement, onsite information management, communication and collaboration between partners, and change and claim management (as presented in Section 2.3.2).

Objective Three: To Explore the Range of Construction Project Information and Identify How These can Best be Managed in a Semantic Web Environment

To achieve this objective, the construction information management concept and traditional information management approaches were reviewed, which are presented in Sections 1.2, 3.2 and 3.3. The challenges faced by traditional information management systems were then highlighted in Sections 1.3 and 3.4. The limitations of current construction information management systems were concluded into three points:

1) Lack of content-based information management;

2) Lack of interoperability of information sources;

3) Lack of context-awareness for Web Services.

As analysed in Section 1.3 and 3.5, the above limitations can be addressed by the use of the Semantic Web. The review of current Semantic Web-related studies in the construction industry revealed that existing research provided fragmented codes and sporadic ontologies, which were independently developed with a lack of interoperation (as reviewed in Section 2.3.2).

On the basis of the identified limitations of traditional construction information management systems and existing Semantic Web-related studies, it is asserted that a systematic study on the Semantic Web-based information management system which integrates existing research outputs is needed. The system should also be able to merge diverse construction information sources and management tools with the overall Semantic Web environment. SWIMS (Semantic Web-based Information Management System for construction projects) is developed for this purpose. The framework of SWIMS is presented in Section 5.2. The framework comprises of two major layers and five sub-layers, which covers all major components within the Semantic Web environment. The Semantic Web-related components are the focus of this research.

Within the Semantic Web environment, construction information sources should be classified into a taxonomy, which involves all terms about construction information

sources. Existing construction information types, classification systems and taxonomies are reviewed and analysed in Section 3.6. The Industry Foundation Classes (IFC) is identified as the most widely adopted information classification system for developing IS and software tools in the construction industry. Its entity schema highly matches with the information representation structure on the Semantic Web. Based on IFC, the SWIMS taxonomy was established, which also combined with other existing taxonomies, such as the process information taxonomy derived from the Process Protocol and specific terms adopted from UniClass facets, as presented in Section 3.6.5. The SWIMS taxonomy comprised terms that define real-world information sources and their properties and relations.

Objective Four: To Implement a System Framework and Its Middleware for Managing Construction Project Information in a Semantic Web Environment

Based on the system framework proposed for managing construction project information in the Semantic Web environment, the semantic components (middleware) and mechanisms involved in the system were developed. This involved three steps:

1) Propose implementation scenario: The documented information management scenarios were presented in Section 5.3;

2) Present models and mechanisms for applying the IFC-based taxonomy in annotating construction information sources: The information annotation models were presented in Sections 5.4.2 and 5.4.3;

3) Identify and specify the interdependencies of information sources within the scenarios: A construction knowledge base, including ontologies and associated instances (as explained in Section 6.5), were developed for this purpose. The instances represent information sources and the ontologies represent the features of the information sources in computer processable languages, which were presented in Section 6.5.

Eventually, in the documented information management scenarios, the construction information sources could be annotated by the first version SWIMS Ontology. The SWIMS Ontology involves a shared ontology for generic information used by all project partners in all domains; a product speciality ontology for construction

product information (building-related products in this case) used by all project partners; and several speciality ontologies used by various project partners in the proposed scenarios.

Objective Five: To Implement the Prototype System Using Practical Project Information and Evaluate the System Demonstrator

The SWIMS implementation process was depicted in IDEF0 diagrams, which described the information management tasks in three modules. These tasks were simulated by applying the prototype annotations and ontologies to 64 practical project documents and 29 project partners' profiles in Protégé (as demonstrated in Chapter 7). A deployment demonstrator was generated, which demonstrated the following system functionality:

1) Content-based Information Management:

- The documented construction information is defined by SWIMS Ontology 1.0. Using ontologies with deductive rules, computers process information and RFI based on their meaning. This supported various Web Services and optimised responses to meet the user's request.
- Synonyms are mapped by computers via ontology merging in SWIMS. Technology standards are also convertible. These lead to the attribute-based and knowledge-based document searches.
- The content of documents is accessible without the need for humans to read the documents.

2) Context-aware Information Services:

- Computers will consider the context of project partners and provide the most appropriate information to users. In the demonstrated deployment, the user's context focused on their role, preferences, and relevant project life-cycle stage.
- An updating alert can be sent to relevant project partners based on the content of information and the role and preference of partners.

3) Interoperable Information Management:

- The information sharing and inter-operation between project partners (and agents) is undertaken by computers via merging speciality ontologies. The

sharing is based on the meaning of information in partners' information repositories, rather than simply document delivery.

- A reliable set of links between documents in diverse repositories is built in SWIMS. The links reflect the relationships of documented information and will not be affected by how project partners customise their document management activities. The content of documents is kept consistent throughout all repositories.

The applicability of the developed system model and ontologies were validated via evaluating the system framework and implementation demonstrator by relevant experts and construction practisers (as presented in Chapter 8). The feedback from respondents verified the rationality of the concept of Semantic Web-based information management for construction projects. The evaluation result revealed the key features of the system, the advantages and limitations of the system, and enablers and barriers for adopting the system in industry. The analysis of the responses also indicated the tasks to be carried out in the next stage.

This thesis presented several potential uses of the Semantic Web in construction information management in Section 2.3.2 and specified application scenarios for the research in Section 5.3. These applications were supported by a system which semantically managed information in the construction sector. The system framework could be found in Section 5.2. In this framework, information is ontologically annotated so as to be automatically processed for completing specific tasks. The annotation model and ontology were presented in Section 5.4 and Chapter 6, while Chapter 7 demonstrates the use of the annotation model and ontology to manage practical project documents. Chapter 8 analyses the evaluation of the system framework and demonstrator. The feedback provided valuable recommendations for further research.

9.3 CONCLUSIONS

The following conclusions can be drawn from the research presented in this thesis:

- The SWIMS framework presented a generic platform which demonstrates the use of the Semantic Web as an integral component of the construction project information management infrastructure, by addressing the key issues of

construction information annotation, ontology development, and use of deductive rules.

- The architecture of the SWIMS framework is extensible. SWIMS integrates various end-user applications, which are supported by various intelligent agents performing information management tasks. Each agent contains an ontology and collaborates with other agents by merging their ontologies. This feature enables the system to integrate independently developed speciality applications and ontologies, especially those IFC-compliant ones because SWIMS Ontologies are developed based on the IFC taxonomy.

- The prototype ontologies demonstrated that construction project information can be managed in the extensible SWIMS knowledge base, involving ontologies and instances. The instances represent diverse construction information sources, which can link to each other directly or indirectly via ontologies.

- The feedback gathered during the system evaluation process demonstrated that the Semantic Web technology can improve construction information management in a number of areas, especially the project knowledge management, collaborative design, and communication between partners (refer to Section 8.4.2-(2)-(iii)).

- The Semantic Web provides an innovative approach to managing construction information because it enables the information in construction documents, both structured and un-structured documents, to be interpretable by computers. This can ensure the efficiency and precision of construction information management.

9.4 THE LIMITATIONS OF THIS RESEARCH

Firstly, the research is constrained by the evolving nature of the Semantic Web as it is relatively new. Some techniques, such as the image annotation method, are not standardised, and others, such as the Proof and Trust, have not yet been developed. Therefore, this research cannot demonstrate all the advantages that construction information management can obtain from the Semantic Web.

Secondly, the number of documents used in this research was limited due to the lack of available annotation tools (as explained in Chapter 8) and the difficulty in obtaining a full set of project documents, which are confidentially protected.

Thirdly, the number of evaluators involved in this research was limited due to the fact that there is only a small number of construction participants with sufficient knowledge of the Semantic Web to adequately evaluate the systems developed.

9.5 RECOMMENDATIONS FOR FURTHER WORK

This research developed semantic components of an information system which proved that the Semantic Web can improve the efficiency of construction information management and solve problems existing in current Web-based information management systems. It has also revealed a number of areas for further research and for action by the construction industry. These are discussed below.

9.5.1 Recommendations for Further Research

The key recommendations for further research are based on extending the work presented here to completely cover all aspects of the construction project information management process and other application areas. These are presented at several levels:

(1) Recommendations for Refining SWIMS

While the value of SWIMS was demonstrated in this research, there is scope for further development along the following lines:

- ***Validation of the Proposed Annotation Mechanisms***

A document-concept-keywords interactive annotation mechanism was initially proposed in the research (see Sections 5.4.2-(2)-(ii) and 7.3.3-(2)-(i)). However, the mechanism was not validated because its validation required developing annotation software and end-user applications, which were not within the scope of the research. Nevertheless, as a part of the whole system, it is necessary to develop project document annotation software which employs the mechanism in automatically annotating project documents and updating the related ontology. If the mechanism is not appropriate to the system, refining it or proposing a new mechanism is necessary.

- ***Further Investigation of Annotation Models for Other Construction Information Sources***

Two annotation models (metafiles and user profiles) were developed. Considering the fragmented nature of construction projects, more effort should be put into annotating other construction information sources, such as the structural elements, materials and equipments. These annotations can use the annotation model developed in Section 5.4 with specific annotation items associated with respective information sources.

It is also necessary to enrich the metadata for more information management scenarios. For example, the annotation to a drawing for a contractor to use on the construction site would include more details about the specific structural elements in that drawing (such as the time frame to install the elements) than that used in the general documented information management scenario.

- ***Enrichment of the SWIMS Ontology Library***

The SWIMS Ontology 1.0 was designed for the documented information management scenarios. It is based on a scan of source documents and it is necessary to enrich it with more detailed ontology elements so as to meet the requirements of representing heterogeneous construction documents.

Besides the document management scenario, there are several other application areas in the construction industry (such as the three suitable applications identified in the system evaluation in Section 8.4.2-(2)-(iii)).These scenarios may have different ontologies. A construction task might employ a single ontology or a combination of various ontologies. It is necessary to collect those common ontologies into an ontology library so as to save user time in ontology development.

- ***Development of Mechanisms for Ontology Maintenance***

During the system evaluation process, evaluators indicated that the ontology maintenance is a serious concern for applying the system in the construction industry. Who, where and how to maintain the ontologies are big challenges for the Semantic Web-based Information System. These problems involve both technical and organisational issues that need to be addressed before the industry can really adopt SWIMS.

- ***Development of Complementary End-user Applications***

This research is essentially a proof-of-concept demonstration which focused on developing the semantic components of the system. It did not address the end-user applications. In further research, it is necessary to develop end-user applications as the interface for delivering the construction information (e.g. sending the project partners' RFIs to the semantic components of the system and returning the managed information to the project partners). End-user applications can be designed for various purposes, such as sharing knowledge between project partners, delivering documents to specific partners, generating real-time project reports, and tracking real-time information. A good end-user application should have friendly interface, be easy to use, compatible to existing databases, and interoperable with other applications. These were identified as the key enablers for adopting SWIMS in industry by the evaluators.

- ***Development of Proof and Trust Functional Modules***

The proof and trust functional modules were not developed in SWIMS 1.0 because of the lack of available techniques. However, these functions are very useful in the construction context. For instance, in material and services procurement and quality/quantity inspection, the explanation of why certain information is inferred and the rating of information reliability are important. The corresponding functional modules should be developed when the related Semantic Web techniques are available.

- ***Undertake Industrial Field Work and Case Studies***

The SWIMS system is an early step towards Semantic Web-based information management in construction and so does not cover all construction project information. It is necessary to apply the system to real projects for capturing practical construction information into the system and evaluating the prototype. There are two methods that could be used for this purpose.

- Field work, in which developers apply the prototype system in real-world projects, is a good approach to test and enrich the developed ontologies and annotations models.

- Case studies, in which evaluators use the prototype system in their projects, are a good method to evaluate and improve the performance of a prototype system. The case studies can further demonstrate the suitability and scalability of the technology for the construction industry and increase the industry awareness of the potential benefits.

(2) Recommendations for Developing the Semantic Web and Its Tools

The Semantic Web is under active development. Some of the findings from this research can inform further development of the Semantic Web, as outlined below.

- ***Development of Ontology Merging Tools and Mechanisms***

This research adopted PROMPT to merge lightweight ontologies. It was discovered that ontology merging is a semi-automatic process. The sequence of merging ontology elements involves a concept, related properties, the next concept, and so on. Conflicts often appear when the properties of a mapped concept are related to other unmapped concepts. Resolving these conflicts is not easy and it is a manual process. When the ontologies become big, especially when the number of intertwined concepts and properties increases, ontology merging takes a considerably long time and the result becomes less reliable. Based on the experience drawn from the research, alternative ontology elements' merging sequence is needed, such as merging all concepts first, then merging properties of each concept, and then configuring conflicts. It would be better if the ontology engineer could choose the appropriate sequence by himself/herself.

- ***Development of Information Extractors and Annotation Tools***

The documented information was manually extracted and annotated into metafiles and user profiles in this research. Existing information extractors and annotation software proved less useful in the following respects:

- Most of these can only be applied to textual files while construction projects involve large volumes of drawings, sketches, tables, pictures, videos and information in other forms;
- Theses were not designed for construction project documents, and so lack pre-specified annotation items for information in the construction context.

Existing information extractors and annotation tools cannot satisfy the need for extracting heterogeneous and multi-media information from diverse construction information sources. Manual annotation is definitely a time-consuming process and will offset the convenience of using these annotations. Software which automatically extracts construction information from various project documents is a prerequisite for applying the Semantic Web technology in the construction industry. This need can be achieved by the development of machine learning techniques and other advanced annotation tools.

- *Development of Ontology Editors and Reasoners*

The research adopted Protégé-OWL 3.2 Bata to develop ontologies and RacerPro to infer deductive rules. It was found that Protégé uses DIG (a standard XML-based language) to communicate with DL (Description Logic) reasoners (such as RacerPro). The current version of DIG (DIG 1.1) has some limitations in the processing of datatype properties. In order to reason with the rest of the ontology, Protégé-OWL filters out datatype properties and logs a warning to users. In the SWIMS Ontology 1.0, datatype properties have to be excluded from the implementation demonstrator. However, much construction information management activities involve datatype properties, such as the conversion of different technical standards and the annotation of a structural element's dimension. A new version of DIG addressing this issue is needed.

9.5.2 Recommendations for the Construction Industry

The key recommendations for industry deployment are as follows:

- *Achieve a Commonly Agreed Taxonomy in the Construction Industry*

During the system evaluation process, evaluators indicated that the use of a common taxonomy should be concerned in applying the system to industry. Theoretically, various ontologies are allowed to be merged despite of their taxonomy as long as they are developed for the same domain. Practically, the similarity of taxonomies significantly enhances the efficiency of ontology merging. Otherwise, much time is need for humans to look for mapped elements of the ontologies. A commonly agreed taxonomy is very useful in applying the system in real-world projects. The review of current construction information classification

systems showed that IFC could be such a taxonomy. Although it is a trend to use IFC in developing new software applications and ontologies, much effort should be put on integrating the IFC with other existing taxonomies and urging the project partners to use a common taxonomy.

- *Study the Economic Issue for Use the System in Industry*

The economic issue was not addressed in this research because the developed system is far from maturity. However, it is necessary to conduct benefit/cost analysis of applying the system when it is ready to be used by industry because even if SWIMS can bring many benefits to the industry, the use of a system are also constrained by some other issues, like cost for IT resources (e.g. hardware, intranet system, server, and staff time for maintenance).

- *Training of Employees to Use the System in Industry*

The training of staffs to use the system is identified as important for its industrial deployment in the system evaluation process. According to the research findings, the use of OWL reasoner might be a key task of training. It is necessary to explore the best approach to training staff.

9.6 CLOSING REMARKS

The successful adoption of the Semantic Web-based Information Management System should considerably improve the efficiency of construction information management. The research in this thesis demonstrated how the system improves construction project information management in the Semantic Web environment. The extensible architecture for the system framework and semantic components (i.e. ontologies) enables the system to integrate independent construction information sources, ontologies and applications, so as to semantically manipulate heterogeneous construction information in a complex construction context. The functions of the system include content-based information management, context-specific information services, and interoperation between different information sources. These functions could facilitate collaboration between project partners by making it easier to access appropriate and timely information from diverse sources, which traditional information management systems cannot achieve. The system promises to enhance the quality of construction activities in a broad area, especially

those information-intensive actions, such as collaborative design, project knowledge management, and communication between partners.

REFERENCE:

4Project (2001) Accessed via WWW. http://www.4projects.com.

Ackoff, R.L. (1989) From Data to Wisdom. *Journal of Applies Systems Analysis*, 16: 3-9.

Albert, S. (1998) Knowledge Management: Living Up to the Hype? *Midrange System*, 11(no.13): 52.

Amara, R. and Salanik, G. (1972) Forecasting: From Conjectural Art toward Science. *Technological Forecasting and Social Change*, 3 (no. 3): 415-426.

Annotea Project (2005) Accessed via WWW. http://www.w3.org/2001/Annotea/.

Anthes, G.H. (1991) A Step Beyond a Database. *Computerworld*, 25(no. 9): 28.

Antoniou, G. and van Harmelen, F. (2004) *A Semantic Web Primer*. Cambridge, Massachusetts: MIT Press.

Anumba, C.J., Ruikar, D., Aziz, Z., Carrillo, P.M. and Bouchlaghem, N. M. (2005) "Semantic Web and Wireless Services for Construction Collaboration". In Soibelman, L. and Pena-Mora, F. eds. *Proceeding of ASCE International Conference on Computing in Civil Engineering*. Cancun [CD-ROM].

Anumba, C., Ruikar, S., Aziz, Z., Carrillo, P.M. and Bouchlaghem, D. (2003) "Towards a Web of Construction Knowledge and Services". In Flood, I. ed. *Towards a Vision for Information Technology in Civil Engineering: Proceeding of the Fourth Joint International Symposium on Information Technology in Civil Engineering*. Nashville [CD-ROM].

Ascher, W. (1979) *Forecasting: An Appraisal for Policymakers and Planners (rev. Ed.)*. Baltimore: Johns Hopkins University Press.

Asite (2000) "Asite – Delivering data logistics". Accessed via WWW. http://www.asite.com.

Aspinall, D. (2004) *An Introduction to Information Management.* British Standards Institution.

Avison, D.E. and Fitzgerald, G. (1991) Information Systems Practice Education and Research. *Journal of Information Systems*, 1 (no. 1): 5-17.

Aziz, Z. (2005) *Context-aware Information Delivery for Mobile Construction Workers*. Ph.D. Thesis, Loughborough University.

Aziz, Z., Anumba, C.J., and Law, K. (2006) "Using Context-awareness and Web-services to Enhance Construction Collaboration". In *Proceeding of Joint International Conference on Computing and Decision Making in Civil and Building Engineering*. Montreal, [CD-ROM].

Aziz, Z., Anumba, C.J., Ruikar, D., Carrillo, P.M. and Bouchlaghem, D. (2004) Semantic Web Based Services for Intelligent Mobile Construction Collaboration. *ITcon,* 9 (Special Issue Mobile Computing in Construction): 367-379.

Barresi, S., Rezgui, Y., Lima, C. and Meziane, F. (2005) "Architecture to Support Semantic Resources Interoperability" In *Proceeding of the First International Workshop on Interoperability of Heterogeneous Information Systems*, New York: Germany ACM Press, pp 79 – 82.

Baskerville, R. and Wood-Harper, A.T. (1998) Diversity in Information Systems Action Research Methods. *European Journal of Information Systems*, 7: 90-107.

bcXML (2003) "eConstruction: eCommerce and eBusiness in the European Building and Construction Industry: Exploiting the Next Generation Internet". Accessed via WWW. http://www.bcxml.org/.

Bechhofer, S., Horrocks, I., Goble, C. and Stevens, R. (2001) "OilEd: a Reason-able Ontology Editor for the Semantic Web". In *Proceeding of Joint*

German/Austrian conference on Artificial Intelligence, KI2001, Vienna: Springer-Verlag pp 396-408.

Benatallah, B., Hacid, M.S., Paik, H.Y., Rey, C. and Toumani, F. (2006) Towards Semantic-driven, Flexible and Scalable Framework for Peering and Querying E-catalog Communities. *Information Systems*, 31 (no. 4-5): 266-294.

Bernaras, A., Laresgoiti, I. and Corera, J. (1996) "Building and Reusing Ontologies for Electrical Network Applications". In *Proceeding of the European Conference on Artificial Intelligence (ECAI'96).* ECAI, pp 298-302.

Bernard, H.R. (1995) *Research Methods in Anthropology, Second Edition.* London: Sage Publications.

Berners-Lee, T. (1998) "Semantic Web Road Map". Accessed via WWW. http://www.w3.org/DesignIssues/Semantic.html.

Berners-Lee, T. (2002) "The Semantic Web". Accessed via WWW. http://www.w3.org/2002/Talks/04-sweb/.

Berners-Lee, T., Hendler, J. and Lassila, O. (2001) The Semantic Web. *Scientific American*, 284(no. 5): 34-43.

Berners-Lee, T. and Connolly, D. (2003) "Semantic Web Application Platform". Accessed via WWW. http://www.w3.org/2000/10/swap/.

BIW Technologies Ltd (2001) Accessed via WWW. http://www.biw.co.uk/home.asp.

Boley, H., Grosof, B., and Tabet, S. (2005) "RuleML Tutorial". Accessed via WWW. http://www.ruleml.org/papers/tutorial-ruleml-20050513.html.

Bouthillier, F. and Shearer, K. (2002) Understanding Knowledge Management and Information Management: the Need for an Empirical Perspective. *Information Research*, 8 (no.1): 141.

Brown, D., Leal, D., McMahon, C., Crossland, R. and Devlukia, J. (2004) A Web-enabled Virtual Repository for Supporting Distributed Automotive Component Development. *Advanced Engineering Informatics*, 18 (no. 3): 173-190.

Bryman, A. and Bell, E. (2003) *Business Research Method.* Oxford: Oxford University Press.

Burstein, F. and Gregor, S. (1999) "The Systems Development or Engineering Approach to Research in Information Systems: an Action Research Perspective". In *Proceeding of 10th Australasian Conference on Information Systems. Wellington,* New Zealand, pp 1-3.

Cai, Q.Y. and Ng, F.F. (2004) "Managing Changes in the AEC Industry-How Can Ontologies help?" In Dikbas A. and Scherer R eds. *eWork and eBusiness in Architecture, Engineering and Construction, Proceeding of The 5th European Conference on Product and Process Modeling in The Building and Construction Industry – ECPPM 2004.* Istanbul: AA Balkema Publishers, pp 305.

Cecez-Kecmanovic, D. (1994) "Engineering Type Information Systems Research: a Discussion on Its Position and Quality". In *Proceeding of the 5th Australian Information Systems Conference*, Caulfield, Vic., Monash Department of Information Systems, pp 767-770.

Chambers, R.L. and Skinner, C.J. (2003) *Analysis of Survey Data*, Chichester: John Wiley & Sons.

Chan, S. and Leung, N. (2004) Prototype Web-Based Construction Project Management System. *Journal of Construction Engineering and Management,* 130 (no. 6): 935-943.

Creswell, J.W. (1997) *Qualitative Inquiry and Research Design: Choosing among the Five Traditions*, CA: Sage Publications.

Christiansson, P. (2003) "Next Generation Knowledge Management System for the Construction Industry". In *CIB W78 Proceeding Construction IT Bridging the Distance*, Auckland: CIB Publication 284, pp 80-87.

Construction Project Information Committee (CPIC) (1997) Crawford, M., Cann, J. and O'Leary, R. eds. *Uniclass (Unified Classification for the Construction Industry)*, London: Royal Institute of British Architects Publications.

CO-ODE (2006) "CO-ODE/HyOntUse". Accessed via WWW. http://www.co-ode.org/.

Copernic Summarizer (2006) Accessed via WWW. http://www.copernic.com/en/products/summarizer/.

Cornish, E. (1977) *The Study of the Future*, Washington, D.C.: World Future Society, pp 108.

Cross, R. (1998) Managing for Knowledge: Managing for Growth. *Knowledge Management*, 1(no. 3):11.

Cycorp, Inc. (2006) "Cycorp". Accessed via WWW. http://www.cyc.com.

Danso-Amoako, M.O., Issa, R. and Cox, R. (2006) "Developing a Framework to Support Data Exchange from Heterogeneous Sources via IFC and Web Services" In *Proceeding of Joint International Conference on Computing and Decision Making in Civil and Building Engineering*. Montreal [CD-ROM].

DARPA (2006) "DAML: The DARPA Agent Markup Language Homepage". Accessed via WWW. http://www.daml.org.

Davies, L.J. and Nielsen, S. (1992) "An Ethnographic Study of Configuration Management and Documentation Practices in an Information Technology Centre". In Kendall, K.E., Lyytinen, K. and De Gross, J. eds. *The Impact of Computer Supported Technology on Information Systems Development*, Amsterdam: Elsevier.

Davies, J., Duke, A. and Stonkus, A. (2003) "Chapter 10: OntoShare: Evolving Ontologies in a Knowledge Sharing System". In Davies, J., Fensel, D. and van Harmelen, F. eds. *Towards the Semantic Web: Ontology-driven Knowledge Management*. Chichester: John Wiley and Sons, pp 161-176.

de Bruijn, J., Ehrig, M., Feier, C., Martín-Recuerda, F., Scharffe, F. and Weiten M. (2006) "Ontology Mediation, Merging and Aligning". In Davies, J., Warren, P. and Studer, R. eds. *Semantic Web Technologies*. Chichester: John Wiley and Sons.

Denny, M. (2004) "Ontology Tools Survey, Revisited". Accessed via WWW. http://xml.com/pub/a/2004/07/14/onto.html.

Ding, Y., Fensel, D., Klein, M. and Omelayenko, B. (2002) The Semantic Web: Yet Another Hip? *Data & Knowledge Engineering,* 41(no. 2-3): 205-227.

Dogac, A., Laleci, G.B., Kirbas, S., Kabak, Y., Sinir, S.S., Yildiz A. and Gurcan Y., (2006) Artemis: Deploying Semantically Enriched Web Services in the Healthcare Domain. *Information Systems*, 31(no. 4-5): 321-339.

Domingue, J., and Motta, E. (2000) Planet-Onto: from News Publishing to Integrated Knowledge Management Support. *IEEE Intelligent Systems*, 15 (no. 3): 2000, page 26-32.

Dowman, M., Tablan, V., Cunningham, H. and Popov, B. (2005) "Web-Assisted Annotation, Semantic Indexing and Search of Television and Radio News". In *Proceeding of 14th International World Wide Web Conference (WWW 2005)*. Chiba, Japan.

Drake, P., Shanks, G. and Broadbent, M. (1998) Successfully Completing Case Study Research: Combining Rigour, Relevance and Pragmatism. *Information Systems Journal*, 8 (no. 4): 273-289.

Duke, A.K. (2001) *A Telepresence Environment for Concurrent Life-cycle Design and Construction*. Ph.D. Thesis, Loughborough University.

e-COGNOS Consortium (2002) "e-COGNOS Field Trials" Accessed via WWW. http://e-cognos.cstb.fr/Downloads/WP4/e-COGNOS-D4.1.pdf.

Egan, J. (1998) *Rethinking Construction, Report of the Construction Task Force on the Scope for Improving the Quality and Efficiency of the UK Construction Industry.* London: Department of Environment.

Ekholm, A. (2005) ISO 12006-2 AND IFC – Prerequisites for Coordination of Standards for Classification and Interoperability. *ITcon,* 10: 275-289.

El-Gohary, N.M. and El-Diraby, T.A. (2006) "Supporting Stakeholder Collaboration in Infrastructure Projects Using Ontology Merger". In *Proceeding of Joint International Conference on Computing and Decision Making in Civil and Building Engineering.* Montreal [CD-ROM].

El-Diraby, T.A., Lima, C. and Fies, B. (2003) A Process-Oriented Taxonomy for Construction Systems: Toward Semantic Exchange of Project Knowledge. Submitted to *Journal of Computing in Civil Engineering*, ASCE.

El-Diraby, T.A., Lima, C. and Feis B. (2005) Domain Taxonomy for Construction Concepts: Toward a Formal Ontology for Construction Knowledge. *Journal of Computing in Civil Engineering*, 19 (no. 4): 394-406.

El-Diraby, T.A. and Gill, S.M. (2006) A Taxonomy for Construction Terms in Privatized-infrastructure Finance: Supporting Semantic Exchange of Project Risk Information. *Construction Management and Economics,* 24 (no.3): 271-285.

Emmerik, M. (2000) "An End-to-end Solution for Improving the Efficiency of the Building Design and Management Process". Accessed via WWW. http://corporate.bricsnet.com/about/documentation/end2end.pdf.

Engels, R. (2001 a) "CORPORUM-OntoExtract. Ontology Extraction Tool. On-To-Knowledge Deliverable 6". Accessed via WWW. http://www.ontonowledge.org/del.shtml.

Engels, R. (2001 b) "CORPORUM-OntoWrapper. Extraction of Structured Information from Web-Based Resources. On-To-Knowledge Deliverable 7". Accessed via WWW. http://www.ontonowledge.org/del.shtml.

Enterprise Project (1997) "Towards a framework for enterprise modelling and integration". Accessed via WWW. http://www.aiai.ed.ac.uk/project/enterprise/.

Enzer, S. (1971) Delphi and Cross-Impact Techniques: An Effective Combination for Systematic Futures Analysis. *Futures*, 3 (no. 1): 48-61.

eXero Summariser (2005) Accessed via WWW. http://www.exago.co.uk/exerosummariser.htm.

Extractor (2003) Accessed via WWW. http://www.extractor.com/Extractor7Evaluation.htm.

FaCT++ (2006) "OWL: FaCT++". Accessed via WWW. http://owl.man.ac.uk/factplusplus/.

Faraj, I. and Alshawi, M. (2004) Global Project Collaboration in the Construction Industry: Standardization and Integration. *International Journal of Computer Application in Technology*, 20 (no. 1/2/3): 26-41.

Farquhar, A., Fikes, R. and Rice, J. (1997) The Ontolingua Server: A Tool for Collaborative Ontology Construction. *International Journal of Human-Computer Studies*, 46(no. 6): 707-727.

Fellows, R.F. and Liu, A. (1997) *Research Methods for Construction (2nd Edition).* Oxford: Blackwell Science.

Fellows, R.F. and Liu, A. (2003) *Research Methods for Construction (3rd Edition),* Oxford: Blackwell Science.

Fensel, D. (2001) *Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce.* Berlin: Springer-Verlag.

Fernández López, M., Gómez-Pérez, A., Sierra, J.P. and Sierra, A.P. (1999) Building a Chemical Ontology Using Methontology and the Ontology Design Environment. *IEEE Intelligent Systems,* 14 (no.1): 37-46.

Ferreira da Silva, C., Lima, C., Medini, L., Chodous, P., and Zarli, A. (2006) "Enhancing Semantic Interoperability among Semantic Resources for Construction". In *Proceeding of Joint International Conference on Computing and Decision Making in Civil and Building Engineering.* Montreal [CD-ROM].

Fetterman, D.M. (1998) "Ethnography: Step by Step". In *Applied Social Research Methods Series, Volume 17,* Walnut Creek, CA: Sage Publications, Inc.

Foddy, W. (1994) *Constructing Questions for Interviews and Questionnaires: Theory and Practice in Social Research.* Cambridge: Cambridge University Press.

Fox, M.S., Barbuceanu, M. and Gruninger, M. (1995) "An Organisation Ontology for Enterprise Modelling: Preliminary Concepts for Linking Structure and Behaviour". In *Proceeding of Fourth Workshop on Enabling Technologies - Infrastructures for Collaborative Enterprises,* West Virginia University.

Fox, M.S. and Grüninger, M. (1998) Enterprise Modeling. *AI Magazine,* 19 (no. 3): 109-121.

Frits, T., Jeff, S., Rasso, S., Reinout, v.R., Michel, B. and Alain, Z. (2001) "bcXML, an XML Vocabulary for Building and Construction". Accessed via WWW. http://vanrees.org/phd/2001_ecce.doc.

FUNSIEC (2006) "Constructing FUNSIEC OSIECS Model with ONDIL Inference". Accessed via WWW. http://195.83.41.67/ondil/InferenceEngine.

Gallaher, M., O'Connor, A., Bettbarn Jr., J. and Gilday, L. (2004) *Cost Analysis of Inadequate Interoperability in the US Capital Facilities Industry.* NIST: NIST Technical Report GCR 04-867.

Galliers, R.D. (1985) "In Search of a Paradigm for Information Systems Research". In Mumford, E., Hirschheim, R., Fitzgerald, G. and Wood-Harper, A.T. eds. *Research Methods in Information Systems*, Amsterdam: North Holland, pp 281-297.

Galliers, R.D. (1992) "Choosing Information Systems Research Approaches". In Galliers R.D. ed. *Information Systems Research: Issues, Methods and Practical Guidelines*, Oxford: Blackwell Scientific Publications, pp 144-162.

Glenn, J.C. and Gordon T.J. (2003) *Futures Research methodology.* ACUNU Millennium Project.

Gómez-Pérez, A. (1998) "Knowledge Sharing and Reuse". In Liebowitz, J. ed. *Handbook of Applied Expert Systems*. Boca Raton: CRC Press LLC, Chapter 10.1-10.36.

Gómez-Pérez, A., Fernández-López, M. and Corcho, O. (2003) *Ontological Engineering: with Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web*, London: Springer-Verlag, pp 322.

Granovetter, M. (1982) "The Strength of Weak Ties: A Network Theory Revisited". In Marsden, P. and Nan, L. eds. *Social Structure and Network Analysis,* Sage Publications, California.

Grüninger, M. and Fox, M.S. (1994) "The Role of Competency Questions in Enterprise Engineering". In *IFIP WG 5.7 Workshop on Benchmarking.* Trondheim: Theory and Practice.

Hanrahan, R.P. (1995) "The IDEF Process Modeling Methodology". Accessed via WWW. http://www.stsc.hill.af.mil/crosstalk/1995/06/IDEF.asp.

Hartmann, S. (1996) "The World as a Process: Simulations in the Natural and Social Sciences". In Hegselmann, R., Mueller, U. and Troitzsch, K.G. eds.

Modelling and Simulation in the Social Sciences from the Philosophy of Science Point of View, Dordrecht: Kluwer, pp 77-100.

Heijden, V. (1996) *Scenarios: The Art of Strategic Conversation.* Chichester: John Wiley and Sons.

Hendler, J. (2001) Agents and the Semantic Web. *IEEE Intelligent Systems.* 16 (no.2): 30-37.

Henry, G., Baraniuk, R.G. and Kelty, C. (2003) *The Connexions Project: Promoting Open Sharing of Knowledge for Education.* Syllabus.

Hibbard, J. (1997) Ernst and Young Deploys App for Knowledge Management. *Information Week*, July 28 1997, pp 28.

Hill, K.Q. (1978) "Trend Extrapolation". In Fowles, J. ed. *Handbook of Futures Research*, Westport: Connecticut, pp 249.

Hitzler, P., Krötzsch, M., Ehrig, M. and Sure, Y. (2005) "What Is Ontology Merging? – A Category-Theoretical Perspective Using Pushouts". In Shvaiko, P., Euzenat, J., Leger, A., McGuinness, D.L. and Wache, H. eds. *Proceeding of the First International Workshop on Contexts and Ontologies: Theory, Practice and Applications (Cand0). Workshop at the 20th National Conference on Artificial Intelligence, AAAI-05.* Menlo: AAAI Press, pp 104-107.

Hjelm, J. (2001) *Creating the Semantic Web with RDF: Professional Developer's Guide*, Wiley, New York: Chichester.

Hlupic, V., Pouloudi, A. and Rzevski, G. (2002) Towards an Integrated Approach to Knowledge Management: 'Hard', 'Soft', and 'Abstract' Issues. *Knowledge and Process Management*, 9 (no. 2): 90-102.

Hooper, S. (2003) "Guidelines for Email and Web Based Surveys". Accessed via WWW. http://crusader-services.com.

Horrocks, I., Patel-Schneider, P.F., Boley, H., Tabet, S., Grosof, B. and Dean, M. (2004) "SWRL: A Semantic Web Rule Language Combining OWL and RuleML". Accessed via WWW. http://www.w3.org/Submission/SWRL/#2.1.

Howard, H.C., Levitt, R.E., Paulson, B.C., Pohl, J.G. and Tatum, C.B. (1989) Computer Integration: Reducing Fragmentation in the AEC Industry. *Journal of Computing in Civil Engineering*, 3 (no. 1): 18-31.

Hughes, J.A., Randall, D. and Shapiro, D. (1992) "Faltering from Ethnography to Design" In *Proceeding of CSCW '92. ACM 1992 Conference on Computer-Supported Cooperative Work: Sharing Perspectives*, New York: ACM Press, pp 115-123.

Hyvönen, E., Mäkelä, E., Salminen, M., Valo, A., Viljanen, K., Saarela, S., Junnila, M. and Kettula, S. (2004) MuseumFinland - Finnish Museums on the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 3 (no. 2-3): 224-241.

IAI (2002) "aecXML Working Documents". Accessed via WWW. http://www.iai-na.org/aecxml/documents.php.

IAI (2006 a) "IFC/ifcXML Specifications". Accessed via WWW. http://www.iai-international.org/Model/IFC(ifcXML)Specs.html.

IAI (2006 b) "IFC Architecture Diagram". Accessed via WWW. http://www.iai-international.org/Model/R2x3_final/index.htm.

Ilsley, P.J. and Krasemann, K. (2002) "Phenomenological Methodology". Accessed via WWW. www.uta.fi/laitokset/aktk/jatko/Phenomenology.doc.

Institution of Civil Engineers (1996) *Civil Engineering Procedure (5th Edition)*, London: Thomas Telford Ltd.

Inxight SmartDiscovery Analysis Server (2006) Accessed via WWW. http://www.inxight.com/products/smartdiscovery/.

ISO (1992) *ISO-10303-1: International Organization for Standardization, Subcommittee 4-Industrial Data and Global Manufacturing Programming Languages, Product Data Representation and Exchange, Part 1, Overview and Fundamental Principles.* Gaithersburgs: National Institute of Standards and Technology.

ISO (1994 a) *ISO-10303-11: Project Data Representation and Exchange Part 11: Description Methods, The EXPRESS Language Reference Manual,* ISO/IEC.

ISO (1994 b) *ISO-10303-21: Project Data Representation And Exchange Part 21: Implementation Methods, Clear Text Encoding Of The Exchange Structure,* ISO/IEC.

ISO (2001) *ISO 12006-2:2001 Building construction- Organization of information about construction works- Part 2: Framework for classification of information (BS).* Geneva: International Standardization Organization/TC59/SC13/WG2.

ISO (2004) *ISO/DIS 12006-3 Building construction- Organization of information about construction works- Part 3: Framework for Object-oriented Information Exchange- revised version 3.* Geneva: International Standardization Organization/TC59/SC13/WG6.

ISO (2005) "STEP ISO-10303". Accessed via WWW. http://www.steptools.com/library/standard/.

Joinson, A.N. and Reips, U. (2005) Personalized Salutation, Power of Sender and Response Rates to Web-based Surveys. *Computers in Human Behaviour,* In Press.

KACTUS (1996) "Esprit Project 8145" Accessed via WWW. http://hcs.science.uva.nl/projects/NewKACTUS/home.html.

Kalton G. (1983) *Introduction to Survey Sampling,* Sage: Sage University.

Keen, P.G.W. (1987) "MIS Research: Current Status, Trends and Needs". In Buckingham, R.A., Hirschheim, R.A., Land, F.F. and Tully, C.J. eds. *Information Systems Education: Recommendations and Implementation*, Cambridge: Cambridge University Press, pp 1-13.

Kiryakov, A., Popov, B., Terziev, I., Manov, D. and Ognyanoff, D. (2004) Semantic Annotation, Indexing, and Retrieval. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2 (no. 1): 49-79.

Kraft, B. and Nagl, M. (2003) "Semantic Tool Support for Conceptual Design". In Flood, I. eds. *Towards a Vision for Information Technology in Civil Engineering: Proceeding of the Fourth Joint International Symposium on Information Technology in Civil Engineering*, Nashville [CD-ROM].

Kwon, O. and Kim, M. (2004) MyMessage: Case-based Reasoning and Multicriteria Decision Making Techniques for Intelligent Context-aware Message Filtering. *Expert Systems with Applications*, 27 (no. 3): 467-480.

Kwon, O., Choi, S. and Park, G (2005) NAMA: a Context-aware Multi-agent Based Web Service Approach to Proactive Need Identification for Personalized Reminder Systems. *Expert Systems with Applications*, 29(no. 1): 17-32.

Lai, Y. and Carlsen, M. (2004) "An Ontology-driven Approach for Monitoring Collaborative Design Knowledge". In Dikbas, A. and Scherer, R. eds. *eWork and eBusiness in Architecture, Engineering and Construction, Proceeding of The 5th European Conference on Product and Process Modeling in The Building and Construction Industry – ECPPM 2004*. Istanbul: AA Balkema Publishers, pp 311-317.

Lai, Y., Carlsen, M., Christiansson, P. and Svidt, K. (2003) "Semantic-Web Supported Knowledge Management System: an Approach to Enhance Collaborative Building Design". In Flood, I. eds. *Towards a Vision for Information Technology in Civil Engineering: Proceeding of the Fourth*

Joint International Symposium on Information Technology in Civil Engineering, Nashville [CD-ROM].

Land, F. (1993) "The Information System Domain". In Galliers, R. ed. *Information Systems Research: Issues, Methods and Practical Guidelines*, Oxford: Blackwell Scientific Publishers, pp 6-13.

Lau, F. (1997) "A Review on the Use of Action Research in Information Systems Studies". In Liebenau, L.A. and DeGross, J. eds. *Information Systems and Qualitative Research*, London: Chapman and Hall, pp 31-68.

Lee, J. and Goodwin, R. (2006) Ontology Management for Large-scale Enterprise Systems. *Electronic Commerce Research and Applications*, 5 (no.1): 2-15.

Lenat, D.B. and Guha, R.V. (1990) *Building Large Knowledge-Based Systems.* Reading, Mass. Addison-Wesley Publishing Company, Inc.

Lewin, K. (1946) Action Research and Minority Problems. *Journal of Social Issues*, 2: 34-46.

Lewis, D., Feeney, K., Tiropanis, T. and Courtenage, S. (2004) "An Active, Ontology-driven Network Service for Internet Collaboration". In *Proceeding of Semantic Web for Web Communities workshop*. Spain: Valencia.

Lewis, I.M. (1985) *Social Anthropology in Perspective.* Cambridge: Cambridge University Press.

Leung, N., Chan, S. and Issa, R.R.A. (2003) "Meta-Data-Based Collaboration in Construction Project Management". In Flood, I. eds. *Towards a Vision for Information Technology in Civil Engineering: Proceeding of the Fourth Joint International Symposium on Information Technology in Civil Engineering*, Nashville, [CD-ROM].

Leydens, J.A., Moskal, B.M., Pavelich, M.J. (2004) Qualitative Methods Used in the Assessment of Engineering Education. *Journal of Engineering Education*, 93(no.1): 65-72.

Liebich, T. (2001) "Ifcxml Language Binding of EXPRESS. Version 1.02". Accessed via WWW. http://www.iai-international.org/IFCXML/Ifc2x/ifcXML_language_binding_V1-02.pdf.

Likert, R. (1932) A Technique for the Measurement of Attitudes. *Archives of Psychology,* 140: 55.

Lima, C., El Diraby, T., Fies, B., Zarli, A. and Ferneley E. (2003) "The e-COGNOS Project: Current Status and Future Directions of an Ontology-enabled IT Solution Infrastructure Supporting Knowledge Management in Construction", In Molenaar, K.R. and Chinowsky, P.S, eds. *Construction Research Congress: Winds of Change: Integration and Innovation in Construction.* Hawaii: Construction Research Congress, pp 19-21.

Löwnertz, K. (1998) "Change and Exchange – Electronic Document Management in Building Design". Accessed via WWW. http://itc.scix.net/data/works/att/fea8.content.05940.pdf.

Macevičiūtė, E. and Wilson, T.D. (2002) The Development of the Information Management Research Area. *Information Research,* 7 (no.3).

Martino, J. (1983) *Technological Forecasting for Decision Making.* New York: Elsevier Science Publishing Company.

McBride, B., Brickley, D. and Miller, E. (2003) "RDFCore Working Group Implementation Report". Accessed via WWW. http://www.w3.org/2001/sw/RDFCore/20030331-advance.html.

McGuinness, D.L., Fikes, R., Rice, J. and Wilder, S. (2000) "An Environment for Merging and Testing Large Ontologies". In *Proceeding of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000),* Breckenridge, Colorado.

McLean, J.M. (1978) "Simulation Modeling". In Fowles, J. ed. *Handbook of Futures Research,* Westport, Connecticut, pp 329-332.

Mezian, F. and Rezgui, Y. (2004) A Document Management Methodology Based on Similarity Content. *Information Sciences*, 158:15-36.

Miles, J.C., Moore, C.J., Kotb, A.S.M. and Jaberisan, H.A. (2000) End User Evaluation of Engineering Knowledge Based System. *Civil Engineering and Environmental Systems*, 17 (no. 4): 293-317.

Moskal, B.M., Leydens, J.A. and Pavelich, MJ. (2002) Validity, Reliability, and the Assessment of Engineering Education. *Journal of Engineering Education,* 91(no. 3): 351-354.

Mutis, I., Issa, R. and Flood, I. (2005) "Semantic Schemas for Specification Processes in the AEC Domain". In Soibelman, L. and Pena-Mora, F. eds. *Proceeding of ASCE International Conference on Computing in Civil Engineering*. Cancun [CD-ROM].

Myers, M.D. (1999) Investigating Information Systems with Ethnographic Research. *Communication of the AIS,* 2 (no.23): 1-20.

Naoum, S.G. (1998) *Dissertation Research and Writing for Construction Students.* Oxford: Butterworth Heinemann.

National Institute of Standards and Technology (NIST), Computer Systems Laboratory. (1993). *Integration Definition for Function Modeling* (IDEF0). Draft Federal Information Processing Standards Publication (FIPS) 183: Department of Commerce.

Norguet, J.P., Zimányi, E. and Steinberger, R. (2006) "Improving Web Sites with Web Usage Mining, Web Content Mining, and Semantic Analysis" In Wiedermann, J., Tel, G., Pokorn, J., ria Bielikov, M. and Stuller, J. eds. *Proceeding of the 32nd Int. Conf. on Current Trends in Theory and Practice of Computer Science*, Heidelberg: Springer Berlin, pp 430-439.

Noy, N.F. and McGuinness, D.L. (2001) "Ontology Development 101: A Guide to Creating Your First Ontology". Accessed via WWW. http://ksl.stanford.edu/KSL_Abstracts/KSL-01-05.html.

Noy, N.F. and Musen, M.S. (2001) "Anchor-PROMPT: Using Non-local Context for Semantic Matching". In *Workshop on Ontologies and Information Sharing at the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI-2001)*, Seattle, pp 63–70.

Noy, N.F. and Musen, M.S. (2003) PROMPT Suit: Interactive Tools for Ontology Merging and Mapping. *International Journal of Human-Computer Studies,* 59 (no. 6): 983-1024.

Noy, N.F. and Musen, M.S. (2004) Ontology Versioning in an Ontology Management Framework. *IEEE Intelligent Systems*, 19 (no. 4): 6-13.

Nunamaker, J. and Chen, M. (1990-91) "Systems Development in Information Systems Research" In *IEEE*, pp 631-639.

O'Brien, R. (2001) "An Overview of the Methodological Approach of Action Research". In Richardson, R. ed. *Theory and Practice of Action Research.* João Pessoa: Universidade Federal da Paraíba.

OilEd (2002) Accessed via WWW. http://oiled.man.ac.uk/.

On-To-Knowledge (2002) "On-To-Knowledge". Accessed via WWW. http://www.ontoknowledge.org/.

Ontolingua (2003) "Stanford KSL Network Services". Accessed via WWW. http://www-ksl-svc.stanford.edu/.

Ontoprise (2006) "Ontobroke: The Power of Inferencing". Accessed via WWW. http://www.ontoprise.de/content/e1171/e1231/index_eng.html.

OntoWeb (2003) "OntoWeb: Ontology-based Information Exchange for Knowledge Management and Electronic Commerce". Accessed via WWW. http://www.ontoweb.org/.

OpenClinical (2004) "MIAKT". Accessed via WWW. http://www.openclinical.org/prj_miakt.html.

OpenCyc.org (2005) Accessed via WWW. http://www.opencyc.org/.

Otjacques, B., Post, P. and Feltz, F. (2003) "Management of Information Flows during Construction Projects". In *Proceeding of the 20th CIB W78 International Conference on Information Technology for Construction.* Auckland, pp 278-285.

Pan, J., Anumba, C.J. and Ren, Z. (2004) "Potential Application of the Semantic Web in Construction". In Khosrowshahi, F. eds. *Proceeding of the Twentieth Annual ARCOM Conference,* Edinburgh: pp 923-930.

Pellet (2006) "Pellet OWL Reasoner". Accessed via WWW. http://www.mindswap.org/2003/pellet/index.shtml.

Pettigrew, A.M. (1985) "Contextualist Research and the Study of Organizational Change Processes". In Mumford, E., Hirschheim, R., Fitzgerald, G. and Wood-Harper, A.T. eds. *Research Methods in Information,* Amsterdam: North Holland, pp 53-78.

Pinheiro da Silva, P., McGuinness, D.L., and Fikes, R. (2006) A Proof Markup Language for Semantic Web Services. *Information Systems*, 31 (no. 4-5): 381-395.

Process Protocol (2002) "Process Protocol Project Level II". Accessed via WWW. http://www.processprotocol.com.

Protégé (2006) "The Protégé Ontology Editor and knowledge Acquisition System". Accessed via WWW. http://protege.stanford.edu/.

Quintas, P. (2005) "The Nature and Dimension of Knowledge Management", In Anumba, C.J., Egbu, C.O. and Carrillo, P.M. eds. *Knowledge Management in Construction.* Oxford: Blackwells, pp 12.

Racer (2006) "Racer Systems GmbH and Co.KG: Protégé". Accessed via WWW. http://www.racer-systems.com/products/tools/protege.phtml.

Racer Systems GmbH and Co KG (2005) "RacerPro and SWRL". Accessed via WWW. http://www.sts.tu-harburg.de/~r.f.moeller/racer/swrl-paper.pdf.

Rapoport, R.N. (1970) Three Dilemmas in Action Research. *Human Relations*, 23 (no. 4): 499-513.

Ren, Z. (2002) *A Multi-agent Systems Approach to Construction Claims Negotiation*. Ph.D. Thesis, Loughborough University.

Ren, Z., and Anumba, C.J. (2004) Multi-agent Systems in Construction–State of the Art and Prospects. *Automation in Construction*, 13 (no. 3): 421-434.

Robson, C. (2002) *Real World Research (2nd Edition),* Oxford: Blackwell.

Rohrlich, F. (1991) "Computer Simulation in the Physical Sciences". In Fine, A., Forbes, M. and Wessels, L. eds. *PSA 1990, Volume 2,* East Lansing, pp 507-518.

Santos, I.A. and Hernandez-Rodriguez, F. (2000) "STEP Based Application Protocol under UML for a Specific Building Code". In Gonçalves, R., Steiger-Garção, A. and Scherer, R.J. eds. *Product and Process Modelling in Building and Construction: Proceeding of the Third European Conference on Product and Process Modelling in the Building and Related Industries,* Lisbon: A.A. Balkema, pp 247-254.

Saunders, M., Lewis, P. and Thornhill, A. (2003) *Research Methods for Business Students,* Pearson Education Limited.

Sayer, A. (1992) *Method in Social Science: A Realist Approach, 2nd Edition.* London: Routledge.

Schnaars, S. (1989) *Megamistakes: Forecasting and the myth of rapid technological change.* New York: The Free Press.

Schreiber, T., Dubbeldam, B., Wielemaker, J. and Wielinga, B.J. (2001) Ontology-based Photo Annotation. *IEEE Intelligent Systems*, 16 (no. 3): 66-74.

Schwarz, B., Svedin, U. and Wittrock, B. (1982) *Methods in Future Studies.* Boulder, Colorado: Westview Press, pp 20-28.

Shanks, G., Rouse, R. and Arnott, D. (1994) "A Review of Approaches to Research and Scholarship in Information Systems". In *Proceeding of the 4th Australian Conference on Information Systems*, Brisbane: Dept. Of Commerce, University of Queensland, pp 197-208.

Sinope Summarizer (2006) Accessed via WWW. http://www.sinope.info/en/Download.

STABU (2004) "A Common International Construction Language". Accessed via WWW. http://www.stabu-lexicon.com/.

Stanford Medical Informatics (2006) "Protégé 2000". Accessed via WWW. http://protege.stanford.edu/.

Stephens, S., LaVigna, D., DiLascio, M. and Luciano, J. (2006) "Aggregations of Bioinformatics Data Using Semantic Web Technology". Accessed via WWW. http://www.websemanticsjournal.org/ps/pub/showDoc.Fulltext/document.pdf?lang=enanddoc=2006-15andformat=pdfandcompression.

Stuckenschmidt, H. and van Harmelen, F. (2004) *Information Sharing on the Semantic Web*. Springer-Verlag.

Sun, M., Bakis, N. and Aouad, G. (2003) A Collaborative Design Appraisal System for Water Treatment Engineering Project. *Advances in Engineering Software*, 34(no. 11-12): 737-743.

Sure, Y. (2002) "Chapter 2 Methodologies and methods for building ontologies from scratch". In Fernández López, M. ed. *OntoWeb Deliverable 1.4: A*

Survey on Methodologies for Developing Maintaining, Evaluating and Reengineering Ontologies. Accessed via WWW. http://ontoweb.org/About/Deliverables/D1.4-v1.0.pdf.

Sure, Y. and Studer, R. (2001) "OntoEdit. On-To-Knowledge Deliverable 3". Accessed via WWW. http://www.ontoknowledge.org/downl/del3.pdf.

Swartout, B., Ramesh, P., Knight, K. and Russ, T. (1997) "Toward Distributed Use of Large-Scale Ontologies". In *Symposium on Ontological Engineering of AAAI.* Stanford, pp 138-148.

Tanyer, A.M. and Aouad, G. (2005) Moving Beyond the Fourth Dimension With an IFC-based Single Project Database. *Automation in Construction*, 14 (no. 1): 15-32.

TOVE Manual (2006) Accessed via WWW. http://www.eil.utoronto.ca/tove/ontoTOC.html.

Updegrove, A. (2005) "The Semantic Web: an Interview with Tim Berners-Lee". Accessed via WWW. http://www.consortiuminfo.org/bulletins/semanticweb.php#1.

Uschold, M. and Gruninger, M. (1996) Ontologies: Principles, Methods, and Applications. *Knowledge Engineering Review,* 11 (no. 2): 63.

van Leeuwen, J.P and van der Zee, A. (2005) Distributed Object Models for Collaboration in the Construction Industry. *Automation in Construction.* 14 (no.4): 491-499.

van Rees, R. and Tolman, F. (2004) "Semantic Web Technologies Applied to Building Specifications". *In*: Betts M. ed. *CIB W78 Information Technology in Construction (CIB 2004).* Toronto, Ontario, Canada. Accessed via WWW. http://vanrees.org/research/papers/cib2004.doc.

van Rees, R., van Vegchel, W. and Tolman, F. (2004) "Practical Use of the Semantic Web: Lessons Learned and Opportunities Found". In Dikbas, A.

and Scherer, R. eds. *eWork and eBusiness in Architecture, Engineering and Construction, Proceeding of The 5th European Conference on Product and Process Modeling in The Building and Construction Industry – ECPPM 2004*. Istanbul: AA Balkema Publishers, pp 329.

W3C (1999) "Signed PICS Labels". Accessed via WWW. http://www.w3.org/DSig/.

W3C (2004 a) "W3C Recommendation: XML Schema Part 2: Datatypes Second Edition". Accessed via WWW. http://www.w3.org/TR/xmlschema-2/.

W3C (2004 b) "W3C Recommendation: RDF Primer". Accessed via WWW. http://www.w3.org/TR/rdf-primer/.

W3C (2004 c) "OWL Web Ontology Language Guide". Accessed via WWW. http://www.w3.org/TR/owl-guide/.

W3C (2006 a) "XML Signature". Accessed via WWW. http://www.w3.org/Signature/.

W3C (2006 b) "Web Services Activity Statement". Accessed via WWW. http://www.w3.org/2002/ws/Activity.

WebODE (2003) "WebODE Ontology Engineering Platform". Accessed via WWW. http://webode.dia.fi.upm.es/WebODEWeb/index.html.

Welch, H. and Watson, E.S. (1979) "Techniques of Futures Research". In Brodzinski, F.R. ed. *Utilizing Futures Research. No. 6,* San Francisco: Jossey-Bass, pp 1-9.

Wikipedia (2005) "Definition of Extranet" Accessed via WWW. http://en.wikipedia.org/wiki/Extranet.

Wikipedia (2006 a) "Definition of Information Management". Accessed via WWW. http://en.wikipedia.org/wiki/Information_management.

Wikipedia (2006 b) "Definition of Information Technology". Accessed via WWW. http://en.wikipedia.org/wiki/Information_technologies.

Wikipedia (2006 c) "Definition of Longitudinal Study". Accessed via WWW. http://en.wikipedia.org/wiki/Longitudinal_study.

Wikipedia (2006 d) "Definition of Likert Scale". Accessed via WWW. http://en.wikipedia.org/wiki/Likert_scale.

Yin, R.K. (1994) *Case Study Research: Design and Methods (3rd Edition),* California: Beverly Hills.

Zonabend, F. (1992) The Monograph in European Ethnology. *Current Sociology,* 40 (no. 1): 49-60.

PUBLICATIONS ARISING FROM THIS RESEARCH

JOURNAL PAPERS:

Pan, J., Anumba, C.J. Semantic Web-based Information Management for Construction Projects. *Advanced Engineering Informatics*. To be submitted in December.

Pan, J., Anumba, C.J. Building Design Information Exchange in a Semantic Web Environment. To be submitted.

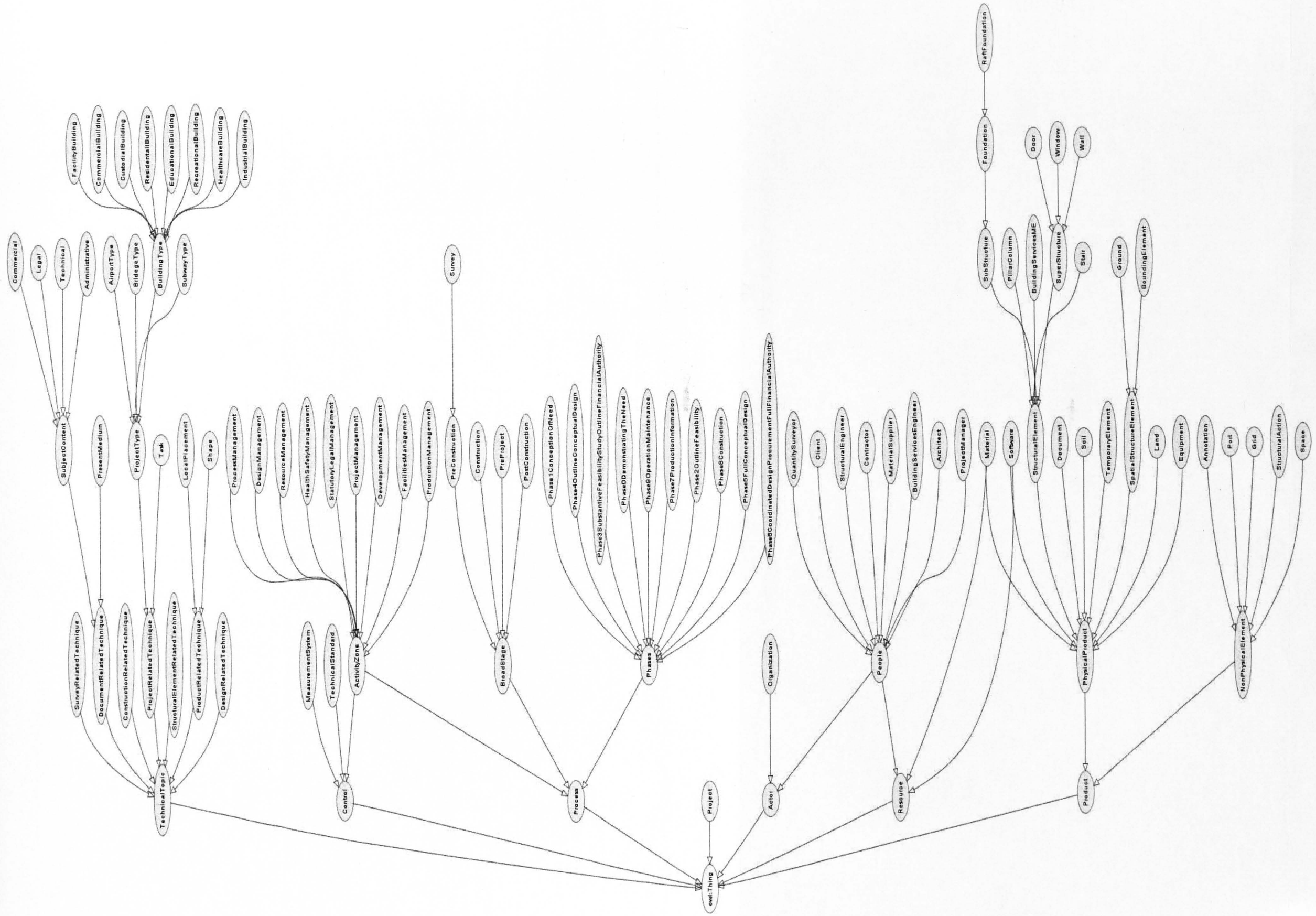
CONFERENCE PAPERS:

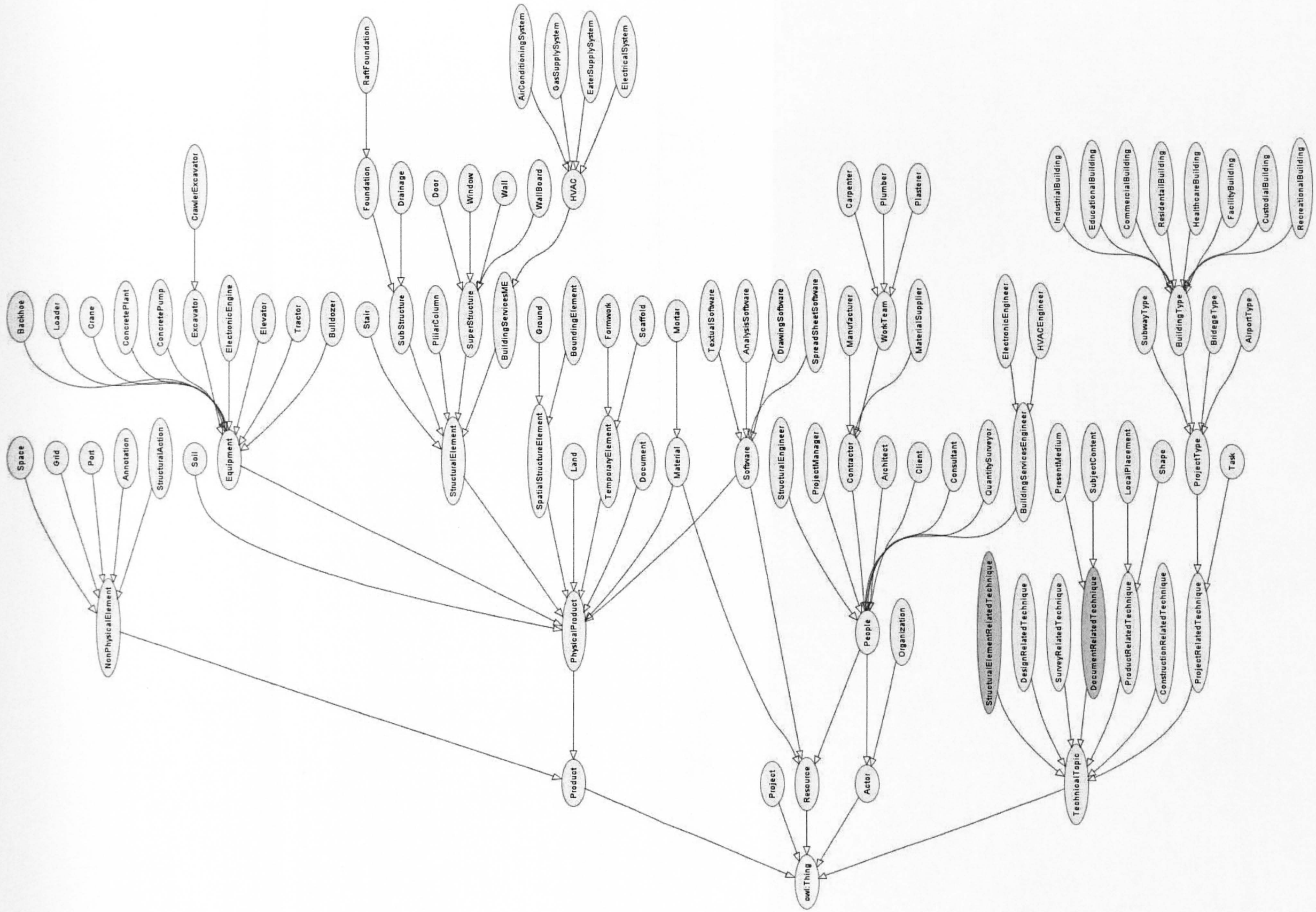
Pan, J., Anumba, C.J. (2006) "Collaborative Design Information Management in a Semantic Web Environment". In *Proceeding of Joint International Conference on Construction Culture, Innovation, and Management (CCIM)*. Dubai, UAE. Accepted for Publication.

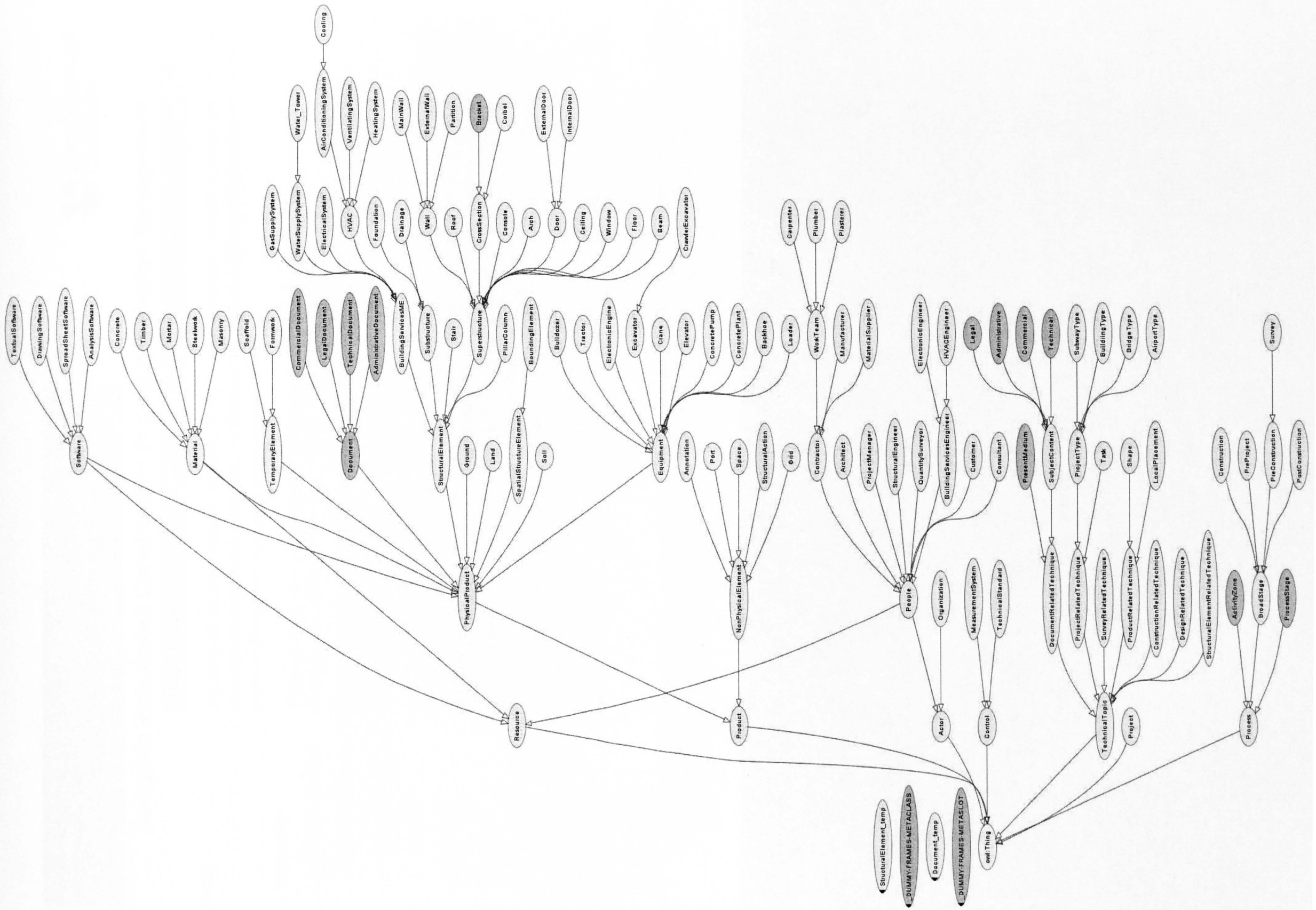
Anumba, C.J., Carrillo, P.M., Pan, J. (2006). "Semantic Web-based Knowledge Management in Construction". In *ASCE and CIB-the 2nd Specialty Conference on Leadership and Management in Construction and Engineering International Perspectives*, Grand Bahama.

Pan, J., Anumba, C.J. (2005) "A Framework for Project Information Management in a Semantic Web Environment". In Soibelman, L and Pena-Mora, F eds. *Proceeding of ASCE International Conference on Computing in Civil Engineering*. Cancun [CD-ROM].

Pan J., Anumba C.J. and Ren Z. (2004) "Potential Application of the Semantic Web in Construction". In Khosrowshahi, F. eds. *Proceeding of the Twentieth Annual ARCOM Conference,* Edinburgh: pp 923-930.


owl:Thing
TechnicalTopic
Control
Process
Project
Actor
Resource
Product
SurveyRelatedTechnique
DocumentRelatedTechnique
ConstructionRelatedTechnique
ProjectRelatedTechnique
StructuralElementRelatedTechnique
ProductRelatedTechnique
DesignRelatedTechnique
MeasurementSystem
TechnicalStandard
ActivityZone
BroadStage
Phases
Organization
People
PhysicalProduct
NonPhysicalElement
SubjectContent
PresentMedium
ProjectType
Task
LocalPlacement
Shape
ProcessManagement
DesignManagement
ResourceManagement
HealthSafetyManagement
StatutoryLegalManagement
ProjectManagement
DevelopmentManagement
FacilitiesManagement
ProductionManagement
PreConstruction
Construction
PreProject
PostConstruction
Phase1ConceptionOfNeed
Phase4OutlineConceptualDesign
Phase3SubstantiveFeasibilityStudyOutlineFinancialAuthority
Phase0DemonstratingTheNeed
Phase9OperationMaintenance
Phase7ProductionInformation
Phase2OutlineFeasibility
Phase8Construction
Phase5FullConceptualDesign
Phase6CoordinatedDesignProcurementFullFinancialAuthority
QuantitySurveyor
Client
StructuralEngineer
Contractor
MaterialSupplier
BuildingServicesEngineer
Architect
ProjectManager
Material
Software
StructuralElement
Document
Soil
TemporaryElement
SpatialStructureElement
Land
Equipment
Annotation
Port
Grid
StructuralAction
Space
Survey
Commercial
Legal
Technical
Administrative
AirportType
BridgeType
BuildingType
SubwayType
FacilityBuilding
CommercialBuilding
CustodialBuilding
ResidentialBuilding
EducationalBuilding
RecreationalBuilding
HealthcareBuilding
IndustrialBuilding
RaftFoundation
Foundation
SubStructure
PillarColumn
BuildingServicesME
SuperStructure
Door
Window
Wall
Stair
Ground
BoundingElement


owl:Thing
Product
Project
Resource
Actor
NonPhysicalElement
PhysicalProduct
People
TechnicalTopic
Space
Grid
Port
Annotation
StructuralAction
Soil
Equipment
StructuralElement
SpatialStructureElement
Land
TemporaryElement
Document
Material
Software
StructuralEngineer
ProjectManager
Contractor
Architect
Client
Consultant
QuantitySurveyor
BuildingServicesEngineer
PresentMedium
SubjectContent
LocalPlacement
Shape
ProjectType
Task
StructuralElementRelatedTechnique
DesignRelatedTechnique
SurveyRelatedTechnique
DocumentRelatedTechnique
ProductRelatedTechnique
ConstructionRelatedTechnique
ProjectRelatedTechnique
Backhoe
Loader
Crane
ConcretePlant
ConcretePump
Excavator
ElectronicEngine
Elevator
Tractor
Bulldozer
CrawlerExcavator
Stair
SubStructure
PliarColumn
SuperStructure
BuildingServicesME
Foundation
Drainage
Door
Window
Wall
WallBoard
HVAC
RaftFoundation
AirConditioningSystem
GasSupplySystem
EaterSupplySystem
ElectricalSystem
Ground
BoundingElement
Formwork
Scaffold
Mortar
TextualSoftware
AnalysisSoftware
DrawingSoftware
SpreadSheetSoftware
Manufacturer
WokTeam
MaterialSupplier
Carpenter
Plumber
Plasterer
ElectronicEngineer
HVACEngineer
SubwayType
BuildingType
BridegeType
AirportType
IndustrialBuilding
EducationalBuilding
CommercialBuilding
ResidentailBuilding
HealthcareBuilding
FacilityBuilding
CustodialBuilding
RecreationalBuilding


Cooling
AirConditioningSystem
Water_Tower
VentilatingSystem
HeatingSystem
MainWall
ExternalWall
Partition
Bracket
Corbel
ExternalDoor
InternalDoor
GasSupplySystem
WaterSupplySystem
ElectricalSystem
HVAC
Foundation
Drainage
Wall
Roof
CrossSection
Console
Arch
Door
Ceiling
Window
Floor
Beam
CrawlerExcavator
Carpenter
Plumber
Plasterer
TextualSoftware
DrawingSoftware
SpreadSheetSoftware
AnalysisSoftware
Concrete
Timber
Mortar
Steelwork
Masonry
Scaffold
Formwork
CommercialDocument
LegalDocument
TechnicalDocument
AdministrativeDocument
BuildingServicesME
Substructure
Stair
Superstructure
PillarColumn
BoundingElement
Bulldozer
Tractor
ElectronicEngine
Excavator
Crane
Elevator
ConcretePump
ConcretePlant
Backhoe
Loader
WorkTeam
Manufacturer
MaterialSupplier
ElectronicEngineer
HVACEngineer
Legal
Administrative
Commercial
Technical
SubwayType
BuildingType
BridgeType
AirportType
Survey
Software
Material
TemporaryElement
Document
StructuralElement
Ground
Land
SpatialStructureElement
Soil
Equipment
Annotation
Port
Space
StructuralAction
Grid
Contractor
Architect
ProjectManager
StructuralEngineer
QuantitySurveyor
BuildingServicesEngineer
Customer
Consultant
PresentMedium
SubjectContent
ProjectType
Task
Shape
LocalPlacement
Construction
PreProject
PreConstruction
PostConstruction
PhysicalProduct
NonPhysicalElement
People
Organization
MeasurementSystem
TechnicalStandard
DocumentRelatedTechnique
ProjectRelatedTechnique
SurveyRelatedTechnique
ProductRelatedTechnique
ConstructionRelatedTechnique
DesignRelatedTechnique
StructuralElementRelatedTechnique
ActivityZone
BroadStage
ProcessStage
Resource
Product
Actor
Control
TechnicalTopic
Project
Process
StructuralElement_temp
_DUMMY-FRAMES-METACLASS
Document_temp
_DUMMY-FRAMES-METASLOT
owl:Thing

Department of Civil & Building Engineering
Loughborough University
Leics, UK, LE11 3TU
Tel: 01509 22 8799

Loughborough University

Semantic Web-based Construction Project Information Management

Questionnaire

Index

1. Introduction
1.1 Problem Definition
1.2 The Semantic Web
1.3 The Framework
1.4 Scenario for Demonstrator
2. Demos
2.1 Introduction to the

This survey will take you no more than 15 minutes. It is designed for evaluating the system for managing construction project information in a Semantic Web environment. Your information will be kept confidential and will not be passed to any third parties. All the responses will be used solely for academic purposes.

Please provide your email address and press *Submit* button when you have finished. If you have any questions, please contact Miss J Pan at J.Pan@lboro.ac.uk.

Please tick options that apply to your situation:

	Yes	No	Don't Know
Which of the following information management tools are you familiar with?			
Traditional information management systems (e.g. manually indexing project documents.)	☐	☐	☐

Ontology

2.2 Ontology Management

2.3 Instance Inference

2.4 Semantic Search

3. Discussion

3.1 Advantages

3.2 Limitations

4. Evaluation

4.1 Questionnaire

Electronic information management systems (e.g. enterprise wide information management system)	☐	☐	☐
Extranets (e.g. Asite, BIW and 4Projects)	☐	☐	☐
Do you think the Semantic Web-based system has advantages over the above information management tools?	☐	☐	☐

Please tick the button that best indicates your opinion on the model and demonstrator:
Rating:1 (Poor), 2(Fair), 3(Satisfactory), 4(Good), 5 (Excellent)

	1	2	3	4	5
How well does the **framework** support construction information management activities?	☐	☐	☐	☐	☐
How compatible is the **framework** with existing construction information management systems?	☐	☐	☐	☐	☐
How well is the content of construction project documents and project partners' information captured in **metafiles** and **user profiles**?	☐	☐	☐	☐	☐
How useful are **metafiles** in saving time to access the information contained in documents?	☐	☐	☐	☐	☐
How well do you think construction project information is represented in the **concept** model?	☐	☐	☐	☐	☐

How well are the relationships between construction information represented by the **concept** hierarchy and **ontologies**?	☐	☐	☐	☐	☐
How useful are **concepts** and the **ontology** in avoiding mistakes during the information exchange process?	☐	☐	☐	☐	☐
How well do you think the flexible and extensible architecture of the **concept** hierarchy and **ontologies** enable information to be shared between diverse communities?	☐	☐	☐	☐	☐
Do you agree that the **concept** hierarchy and **ontologies** are reusable?	☐	☐	☐	☐	☐
How well are the classification and **inference** functions applied to the construction information?	☐	☐	☐	☐	☐
How useful are the classification and **inference** functions in keeping information consistent between the project partners?	☐	☐	☐	☐	☐
How useful are the classification and **inference** functions in distributing timely information?	☐	☐	☐	☐	☐
In general, how useful are classification and **inference** functions in improving the efficiency of construction information management?	☐	☐	☐	☐	☐
How well does the semantic **search** help in understanding the meaning of a request?	☐	☐	☐	☐	☐
How well does the semantic **search** help project partners get appropriate and timely information?	☐	☐	☐	☐	☐

Please rank the ease of use with which the following can be undertaken from 1 (Very Difficult), 2 (Difficult), 3 (Fair), 4 (Easy), 5 (Every Easy).

	1	2	3	4	5
Specifying **concepts** which the information is accompanied with;	☐	☐	☐	☐	☐
Defining a **concept**, e.g. documents, structural elements and actors;	☐	☐	☐	☐	☐
Relating a new concept to existing **concepts**;	☐	☐	☐	☐	☐
Specifying **metafiles** to concepts, e.g. Actors have attributes like name, e-mail, role, and interests;	☐	☐	☐	☐	☐
Editing **ontology** for concepts, e.g. Actors are disjoint with Documents;	☐	☐	☐	☐	☐
Adding **individuals** to concepts, e.g. Mr. Smith is an individual of Actors;	☐	☐	☐	☐	☐
Using OWL reasoners to make a **reasoning**, e.g. consistency checking and inferred ontology computing;	☐	☐	☐	☐	☐
Using semantic **search** service to make a query, e.g. selecting searched range and concepts.	☐	☐	☐	☐	☐

Please tick the top three areas which will get the most benefits from Semantic Web technologies in your opinion.

Information Management and Exchange	
☐	Information collection to facilitate decision making in the pre-design stage, such as feasibility study;
☐	Information sharing between design team members;
☐	Information distribution and collection on the construction site (such as the retrieval of up-to-date drawings);
☐	The collection of information for project inspection and retrieval of defect information;
☐	Information retrieval during building maintenance;
Collaborative Communications	
☐	The communication between project team members;
Procurement	
☐	E-commerce for procurement of goods and services for the project;
Health and Safety Management	
☐	Distributing health and safety documents to related project team members;
Knowledge Management	
☐	Post-project evaluation;

<

- ☐ Knowledge management on a personal basis;
- ☐ Project knowledge management;
- ☐ Organizational knowledge management;

Others, please specify

Do you think the developed construction project information concepts and ontology would be better if more concepts/ontology are added into it? If so, please specify the concepts/ontology you would like to see:

What do you consider the best features of the demonstrator?

5.

In what way(s) can the demonstrator be improved?

Would you consider adopting the system in your organisation?

☐ Yes ☐ No

What would be the enablers for adopting the system?

What would be the barriers to adopting the system?

Please provide the following background information:

Your E-mail address (Required)

Your current occupation: Please state your

Job Title:

Speciality (e.g. architecture, logistics, collaborative working, off-site manufacture and contract management):

How long have you worked in the construction industry?

Submit Reset

Website Maintained by: J.Pan@lboro.ac.uk - LU Home - Accessibility - © Loughborough University