

Building Digital Bridges
Improving digital collaboration through
the principles of Hyperlinked Practice

By

David Henry Harrison

A thesis
submitted to the Victoria University of Wellington
in fulfilment of the requirements for the degree of
Doctor of Philosophy
in Architecture

Victoria University of Wellington

2010

Abstract

Effective collaboration requires access to timely and relevant information, but this is difficult given the complexity of the architectural design process and the segmentation of the architecture, engineering and construction industry. Effective collaboration is further complicated by the quantity and density of the digital information generated within a project, and the irregular adoption of technology by different team members. Consolidating project information within Building Information Models has improved its management, but the technology's complexity limits who can contribute to it. This is a problem, because team members are capable of collaborating more effectively when they can record and reflect upon a comprehensive record of the project's design process.

The aim of research was to identify how information technology can assist architectural project teams to collaborate by more inclusively and comprehensively recording and reflecting upon the design process.

To address this problem, this research proposes that the industry adopt Hyperlinked Practice, which is the creation of a distributed cloud of interconnected information describing an architectural project's events, activities and digital artefacts. A set of fundamental principles were identified that would be used to guide the design and deployment of digital collaboration tools capable of facilitating Hyperlinked Practice. To ensure a flexible and inclusive environment, the principles were derived from concepts proven within the World Wide Web.

To validate these principles, their collaboration influence, potential, and industry applicability was tested within a software prototype utilised in a university architecture course and two thought experiments. The results from testing the software prototype suggest that the principles are capable of influencing collaboration in a manner that promotes the recording of the design process, and reflection upon it. The thought experiments demonstrated that the principles provided an excellent framework for evaluating a digital collaboration tool's ability to facilitate Hyperlinked Practice.

Based on these results, the research concluded the identified principles of Hyperlinked Practice were capable of facilitating a collaboration environment that would allow the design process to be comprehensively recorded and reflected upon.

Acknowledgments

I would like to acknowledge Michael Donn for his supervision, insight and support throughout my studies at Victoria University. Our weekly conversations about the work I had, or had not, undertaken during the week were always interesting and worthwhile.

I would also like to acknowledge Robert Amor and Henry Skates for their valuable and under utilised input during the course of the research.

Many thanks go to my family, Ian, Cheryl and Michael, for their encouragement during this long and winding process.

Finally, I would like to thank my wife Emma for her steadfast belief, love and support throughout this roller-coaster of a ride. During the closing months of the thesis she took on the dual roles of chief editor and cheerleader, and this timely assistance was invaluable. Undoubtedly without her input and unwavering confidence I would not have been able to complete this body of work. I am now looking forward to the day when she begins her own thesis and I can return the favour.

Table of Contents

List of illustrations	8
List of Tables	11
1. Introduction	12
1.1. Problem statement	12
1.2. Aim	13
1.3. Research approach	13
1.4. Thesis structure	13
1.5. Summary	15
2. The Importance of Timely & Relevant Information	16
2.1. Collaboration - the lifeblood of an architectural project	16
2.1.1. What is effective architectural collaboration?	17
2.1.2. Barriers to effective architectural collaboration	19
2.2. Timely and relevant information benefits collaboration	20
2.2.1. Understanding design requirements and decisions	20
2.2.2. Involving relevant parties in design discussion	21
2.3. Challenges communicating timely and relevant information	22
2.3.1. Architecture projects are unique and complex environments	23
2.3.2. Participants do not understand the justification behind decisions	24
2.3.3. Project teams exist within a segmented industry	25
2.3.4. Time extremes restrict and erode project knowledge	27
2.4. Improving collaboration through Building Stories	28
2.4.1. The journey is as important as the destination	29
2.4.2. The benefit of Building Stories to the collaborative design process	32
2.4.3. Using Building Stories to improve timely access to relevant information	34
2.5. Summary	35
3. The Digitally Fragmented Project Team	37
3.1. Architectural collaboration's digital migration	37
3.1.1. The elimination of distance and bandwidth restrictions	38
3.1.2. Immediate and frictionless collaboration interactions	40
3.1.3. The digital downside of Internet-based architectural collaboration	45
3.2. The rise of the Building Information Model	49
3.2.1. The digital model as collaboration hub	50

3.2.2.	What is a Building Information Model?	52
3.2.3.	BIM in Practice	54
3.3.	Digital collaboration barriers in a BIM-centric team	59
3.3.1.	The collaboration shortcomings of BIM	59
3.3.2.	Digital fragmentation and the need for timely and relevant information	63
3.4.	Summary	65
4.	Improving Collaboration Through Hyperlinked Practice	66
4.1.	Putting Building Stories into architectural practice	67
4.1.1.	Recording, reflecting, rendering and recalling	68
4.1.2.	Constructing Building Stories within project teams	69
4.1.3.	Overcoming the barriers to Building Stories	72
4.2.	Constructing Building Stories through Hyperlinked Practice	72
4.2.1.	Embracing the Web to bridge the digital fragmentation of the project team	72
4.2.2.	Adopting Hyperlinked Practice to record Building Stories	74
4.2.3.	The need for fundamental principles of Hyperlinked Practice	76
4.3.	Summary	77
5.	The Principles of Hyperlinked Practice	78
5.1.	Learning from those who shaped the World Wide Web	79
5.2.	Identifying the principles of Hyperlinked Practice	82
5.3.	The influence of the principles on the project information cloud	90
5.4.	Summary	91
6.	Validating the Principles of Hyperlinked Practice	92
6.1.	Avoiding the 'better' collaboration quandary	92
6.2.	Establishing influence, applicability and usefulness	93
6.2.1.	Influence	93
6.2.2.	Applicability	94
6.2.3.	Usefulness	94
6.3.	A review of alternative approaches	94
6.3.1.	Develop and test a software prototype	96
6.3.2.	Comparative case studies of digital collaboration	98
6.3.3.	Thought experiments	99
6.4.	A hybrid software prototype/thought experiment testing strategy	100
6.5.	Summary	101
7.	Testing the Principles within a Software Prototype	102

7.1. The testing environment for the Reasonate prototype	103
7.1.1. Justification of the test environment	103
7.1.2. Composition of the test audience	106
7.1.3. Course objectives and outline	106
7.1.4. Collaboration within the course	107
7.1.5. The role of Reasonate	108
7.1.6. Ethics approval	108
7.2. Testing the influence of each principle	109
7.2.1. Testing the principle of situational awareness	110
7.2.2. Testing the principle of ubiquity	111
7.2.3. Testing the principle of comprehension	112
7.2.4. Testing the principle of context sensitivity	113
7.2.5. Testing the principle of emulative modularity	114
7.2.6. Testing the principle of emotive semantics	115
7.2.7. Testing the principle of decentralisation	116
7.3. The design of the Reasonate prototype	117
7.3.1. The underlying design motives	117
7.3.2. Core functionality	120
7.3.3. The Reasonate use and collaboration influence questionnaire	128
7.4. Constructing the Reasonate prototype	133
7.4.1. A purpose built prototype	133
7.4.2. The influence of agile development	133
7.4.3. The influence of user feedback	135
7.4.4. Independent reimplementations of the prototype	139
7.5. Testing Reasonate	141
7.5.1. Prototype deployment	141
7.5.2. Prototype access	141
7.5.3. Nightly database snapshots	141
7.5.4. Instructions on Reasonate use	142
7.5.5. Timeline of the testing period	143
7.6. Summary	145
8. Reviewing the Performance of the Software Prototype	146
8.1. The successful testing of the Reasonate prototype	147
8.1.1. Reasonate usage statistics	147
8.1.2. Questionnaire submissions	147
8.1.3. The overall influence of Reasonate	148

8.2. The influence of the Hyperlinked Practice principles	149
8.2.1. An overview of each principle's demonstrated influence	149
8.2.2. The effect of situational awareness	151
8.2.3. The effect of ubiquity	154
8.2.4. The effect of comprehension	157
8.2.5. The effect of context sensitivity	159
8.2.6. The effect of emulative modularity	162
8.2.7. The effect of emotive semantics	166
8.2.8. The effect of decentralisation	170
8.3. Discussing the influence of Hyperlinked Practice	172
8.3.1. Reviewing the influence of the principles within Reasonate	173
8.3.2. The Reasonate testing process	183
8.4. Summary	185
9. Exploring the Potential of Hyperlinked Practice	186
9.1. Achieving Hyperlinked Practice by applying its principles	186
9.1.1. Ranking how the principles are embodied within a collaboration tool	187
9.1.2. Visualising how the principles are embodied within a collaboration tool	190
9.1.3. Applying the principles within digital collaboration thought experiments	191
9.2. Using micro-blogging to record architectural design conversation	192
9.2.1. Broadcasting and monitoring simple messages with micro-blogging	192
9.2.2. Integrating the design conversation with the digital model	193
9.2.3. Micro-blogging within a project's information cloud	195
9.2.4. Developing an architectural collaboration micro-blogging platform	198
9.2.5. Summarising architectural micro-blogging	201
9.3. Bluestreak and the birth of a collaboration kernel	201
9.3.1. Seamless collaboration within a fragmented digital environment	202
9.3.2. Using a collaboration kernel to integrate collaboration interactions	206
9.3.3. The untapped potential of Bluestreak	209
9.3.4. A Bluestreak in the project's information cloud	211
9.3.5. Summarising the collaboration kernel	219
9.4. Summary	219
10. Discussion and Summary	220
10.1. Recording the design process through Hyperlinked Practice	221
10.2. Key research findings	224
10.2.1. The identified principles of Hyperlinked Practice are valid	225

10.2.2. Hyperlinked Practice can enable the digital recording of the design process	227
10.3.Applying Hyperlinked Practice within the industry	228
10.4.Limitations of this research	230
10.4.1. The design and testing of the Reasonate software prototype	230
10.4.2. Thought experiments	233
10.4.3. The organisational and legal barriers to Hyperlinked Practice	233
10.5.Future work	234
10.5.1. The isolated validation of each Hyperlinked Practice principle	234
10.5.2. The holistic testing of Hyperlinked Practice within the industry	237
10.6.Summary	240
Appendix	241
Bibliography	243

List of illustrations

Figure 2.1: Diagram illustrating the hypothetical benefit of cooperation, coordination and collaboration to architectural decision making and the effort required to achieve	18
Figure 2.2: The ‘over the wall’ syndrome	26
Figure 2.3: A diagram showing the relationship between effort and effect during the architectural design and construction process	27
Figure 2.4: An illustration of a Building Story’s components and their relationship to each other	31
Figure 2.5: Design as a circular process and its relationship to the events, activities and artefacts within a Building Story	33
Figure 3.1: The use of Knowledge Nodes to support information discovery through the weaving together of project resources	41
Figure 3.2: The relationship between the virtual office and the virtual team	42
Figure 3.3: Organisation levels and adoption barriers	49
Figure 3.4: Collaboration benefits of the digital model	51
Figure 3.5: The effort/time relationship between BIM and traditional digital modelling methods, relative to the cost of design changes	54
Figure 3.6: The adoption of BIM within projects	55
Figure 3.7: The interfaces of a BIM server and the clients which interact with them	58
Figure 3.8: The use of multiple digital models within the project team and the complex information flow which occurs as a result	62
Figure 3.9: The digital fragmentation caused by the technologies used within the project team	64
Figure 4.1: The four phases of Building Story construction and utilisation	69
Figure 4.2: The distributed nature of a project’s information cloud.....	75
Figure 4.3: Constructing a Building Story event within a project’s information cloud.....	76
Figure 7.1: An outline of the BBS303 ‘Digital Craft’ course.....	107
Figure 7.2: The process of developing practical tests & hypothesis for each principle.....	109
Figure 7.3: Reasonate functionality derived from the principles of Hyperlinked Practice	117
Figure 7.4: The relationship between blog posts and files within a conventional blog and one optimised for architectural collaboration	119

Figure 7.5: The sets of tags present within a conventional blog and one optimised for architectural collaboration	120
Figure 7.6: A personal blog within Reasonate.....	121
Figure 7.7: The blog publishing interface within Reasonate.....	122
Figure 7.8: Reasonate’s file information screen showing a file’s different versions	123
Figure 7.9: Reasonate’s advanced search form	124
Figure 7.10: Examples of the RSS icons and text within Reasonate’s interface.....	125
Figure 7.11: The ‘add tag’ and ‘tag listing’ interfaces for a Reasonate blog post	126
Figure 7.12: Reasonate’s tag search screen and personal “tag cloud”	126
Figure 7.13: Reasonate’s questionnaire interface for conducting the user survey	128
Figure 7.14: An overview of Reasonate functionality driven by user feedback	135
Figure 7.15: The Blogger-based reimplementaion of Reasonate	139
Figure 7.16: A timeline of the significant events during Reasonate testing.....	143
Figure 8.1: The influence of Reasonate on the student’s experience of the course.....	148
Figure 8.2: The influence of Reasonate on the team’s ability to work effectively	148
Figure 8.3: A box-plot analysis of the total Reasonate contributions per student	152
Figure 8.4: Reasonate’s influence on the reflection process and student motivation.....	153
Figure 8.5: Activity graphs comparing the times of day contributions occurred.....	153
Figure 8.6: The general locations where students interacted with Reasonate.....	155
Figure 8.7: The student’s perception of their blogging frequency during testing.....	155
Figure 8.8: A breakdown of the digital file formats uploaded to Reasonate over time.....	156
Figure 8.9: The students’ Internet experience prior to the course	157
Figure 8.10: The daily growth and running total of posts published to Reasonate.....	158
Figure 8.11: The daily growth and running total of tags created within Reasonate.....	158
Figure 8.12: A box-plot analysis of the total Reasonate contributions per project	160
Figure 8.13: The breakdown of content contributed to each project ordered by the total number of blog posts.....	161
Figure 8.14: The content contribution activity of four representative project teams	161
Figure 8.15: A comparison of the overall post and tag growth rates during the Reasonate (2006) and Blogger (2007) testing periods	163

Figure 8.16: An ordered comparison of the total contributions to Reasonate and Blogger by individual users.....	164
Figure 8.17: The quantity and length of blog posts contributed to Reasonate (2006)	165
Figure 8.18: The quantity and length of blog posts contributed to Blogger (2007).....	165
Figure 8.19: The students' perception of their tagging habits during testing.....	167
Figure 8.20: The reasons given by students for tagging Reasonate content	167
Figure 8.21: A comparison of the total number of blog posts and tags created per Reasonate and Blogger user	168
Figure 8.22: The overall semantic structure formed by Reasonate users.....	169
Figure 8.23: Examples of student websites published to Reasonate's Web Area.....	171
Figure 8.24: An ordered breakdown of the total amount of content published by each student to Reasonate's blog and Web Area sections	171
Figure 8.25: An ordered breakdown of hyperlinks embedded into student content	172
Figure 8.26: A potential Reasonate interface change to promote tagging.....	180
Figure 8.27: Visualising the evolution of a project's semantic structure over time	181
Figure 9.1: Visualising the ratings assigned to each Hyperlinked Practice principle.....	190
Figure 9.2: The embodiment of Hyperlinked Practice within current collaboration tools.	190
Figure 9.3: Comparing the embodiment of the principles of Hyperlinked Practice within a typical micro-blogging service and an AEC-specific micro-blogging service.....	198
Figure 9.4: Increasing the decentralisation of micro-blogging through the use of project-specific and organisation-specific micro-blogging services	200
Figure 9.5: The digital collaboration interactions within an architectural project team.....	202
Figure 9.6: The Project Bluestreak social networking web interface.....	210
Figure 9.7: Comparing how the principles of Hyperlinked Practice are embodied within the current and potential future implementations of Project Bluestreak	211
Figure 9.8: The three proposed layers of Bluestreak application integration.....	214
Figure 9.9: The collaboration kernel as a hub for integrating third-party applications.....	216
Figure 10.1: Applying an A/B testing strategy to the testing of Reasonate.....	236
Figure 10.2: The design for a distributed software prototype for use within the industry.	239

List of Tables

Table 3.1: The four-stage model of virtual team maturity	45
Table 4.1: Requirements and challenges of Building Story development	70
Table 5.1: Deriving the principle of situational awareness	83
Table 5.2: Deriving the principle of ubiquity	84
Table 5.3: Deriving the principle of comprehension	85
Table 5.4: Deriving the principle of context sensitivity	86
Table 5.5: Deriving the principle of emulative modularity	87
Table 5.6: Deriving the principle of emotive semantics	88
Table 5.7: Deriving the principle of decentralisation	89
Table 6.1: An overview of the characteristics of each potential testing methodology	95
Table 6.2: The embodiment of the principles of Hyperlinked Practice within existing digital collaboration technologies	97
Table 6.3: Ability of the methodology combination to test the validity of the principles ..	100
Table 7.1: Collaboration similarities and differences between a university testing environment and a professional project team	105
Table 7.2: The intention of the survey questions and their contribution to the analysis of the principles of Hyperlinked Practice	129
Table 7.3: A functional comparison between Reasonate (2006) and Blogger (2007)	140
Table 8.1: A summary of each principle's demonstrated influence within Reasonate	149
Table 8.2: An analysis of the content published per student in an average week	151
Table 8.3: A breakdown of the digital files uploaded to Reasonate by format	155
Table 9.1: The rating system for how thoroughly the principles of Hyperlinked Practice are embodied within a digital tool	187
Table 9.2: The purpose, nature and enabling technologies of the digital interactions within an architectural project team	203
Table 9.3: The identity systems used during a typical digital collaboration interaction	205
Table 10.1: The functional differences between the control and alternate prototypes	235

1. Introduction

Bridging the digital collaboration barriers within architectural project teams

1.1. Problem statement

Effective collaboration requires access to timely and relevant information (Emmitt & Gorse, 2003, p. 11), but this is difficult given the complexity of the architectural design process and the segmentation of the architecture, engineering and construction (AEC) industry. (Emmitt & Gorse, 2003, p. 6) The migration to a digital collaboration environment has overcome many traditional communication barriers, because project teams are now able to exchange more information, more frequently over greater distances. Unfortunately, many project teams are now struggling to come to terms with the quantity and density of the digital information generated, and irregular adoption of technology within them has created significant disparities between members. When combined, these digital impediments pose significant challenges that impair the effectiveness of collaboration within the team.

The ability to manage a project's information has been improved by the adoption of the Building Information Model (BIM), which is a comprehensive virtual representation of the proposed architecture. (Aranda-Mena, Crawford, Chevez & Froese, 2008, p. 11) Unfortunately, the complex nature of BIM, coupled with the unequal distribution of information technology knowledge and resources within the industry, means that few members of the project team can directly contribute to the digital model. This lack of accessibility increases the digital fragmentation within the team, and makes it difficult for participants to accurately record their design decisions. This is a problem, because team members are capable of collaborating more effectively when they can reflect upon and understand the project's design process. (Martin, Heylighen & Cavallin, 2005, p. 35) Despite the promising developments on the World Wide Web to enable inclusive digital collaboration, the AEC industry is yet to employ information technology in a manner that allows the design process to be comprehensively recorded and reflected upon by the entire project team.

1.2. Aim

The aim of this thesis is to identify how information technology can assist architectural project teams to collaborate by more inclusively and comprehensively recording and reflecting upon the design process.

1.3. Research approach

To achieve this aim, this research proposes that architectural project teams adopt Hyperlinked Practice, which is the creation of a distributed cloud of interconnected information describing a project's events, activities and artefacts. The fundamental principles of Hyperlinked Practice were derived from the proven concepts of the World Wide Web (Berners-Lee, 1998a), because it is the most successful digital medium for enabling large numbers of people to record, share and reflect upon digital information. The identified principles were validated by demonstrating their ability to influence collaboration behaviour, be applied within the industry, and whether they were capable of facilitating a Hyperlinked Practice collaboration environment. These three characteristics were demonstrated by testing the principles within a software prototype and a series of thought experiments. The results of these tests served to validate the selection of the principles, and provided an indication of how a digital collaboration environment that embodied the principles of Hyperlinked Practice could enable the design process to be more comprehensively recorded and reflected upon by industry project teams.

1.4. Thesis structure

Chapter 2 highlights the benefits and barriers to timely and relevant collaboration information within architectural project teams. Building Stories (Martin, Heylighen & Cavallin, 2003) are identified as a means of improving the project team's ability to access to timely and relevant information, as they promote the comprehensive recording of the design process for later reflection.

Chapter 3 reviews the industry's rapid adoption of digital communication and collaboration processes, and the subsequent information management and design comprehension problems now faced by project teams. The current industry trend of consolidating project information within Building Information Models addresses some of these issues, but the

complexity of this technology restricts the project team's ability to access and contribute to the digital record of the design process.

Chapter 4 puts forward the concept of a Hyperlinked Practice digital collaboration environment, where the entire project team can contribute to an online, distributed and interconnected record of the design process, and collaboratively construct Building Stories based on this cloud of information.

Chapter 5 derives the fundamental principles of Hyperlinked Practice from the proven concepts of the World Wide Web. Individually these principles address specific collaboration challenges, and when collectively embodied within a digital collaboration environment they are intended to facilitate Hyperlinked Practice.

Chapter 6 identifies the most appropriate means of validating the identified principles. To achieve this a software prototype and a series of thought experiments were selected to demonstrate the collaboration influence of the principles, and their ability to facilitate Hyperlinked Practice.

Chapter 7 describes the design of the software prototype and associated tests that were used to demonstrate the collaboration influence of the principles.

Chapter 8 details the results from testing the software prototype within a university course on collaborative digital modelling. The findings and limitations of these results are discussed so to establish the demonstrated collaboration influence of each principle.

Chapter 9 applies the principles within a series of thought experiments to demonstrate their ability to guide the development of a Hyperlinked Practice collaboration environment that enables the design process to be comprehensively recorded and reflected upon.

Chapter 10 concludes the thesis with a summary, and a discussion of the contributions, limitations and future work related to this research.

1.5. Summary

Architectural collaboration is more effective when project teams can comprehensively record and reflect upon the design process. Unfortunately, the digital collaboration environments within the AEC industry are unable to comprehensively or inclusively record the design process. To address this problem, the aim of this thesis is to identify how information technology can assist architectural project teams to collaborate by more inclusively and comprehensively recording and reflecting upon the design process. The next chapter highlights the benefits and barriers to timely and relevant collaboration information, and how understanding the design process can improve access to it.

2. The Importance of Timely & Relevant Information

Using Building Stories to improve the effectiveness of architectural collaboration

Effective collaboration is essential if those involved in an architecture project are to complete the process efficiently and to the identified requirements. Unfortunately the Architecture, Engineering and Construction (AEC) industry has a checkered history when it comes to the creation of cohesive collaboration environments, and as a result, many projects suffer from miscommunication, poor decisions and overlooked requirements. These problems can be avoided when timely and relevant information is disseminated throughout the project team. Timely and relevant information improves the participant's awareness of the current state of the project, the justifications behind design decisions, and the activities of other team members. Hampering this flow of timely and relevant information is the complexity and lack of understanding within project teams, brought on by industry segmentation and time pressures. To overcome these barriers, project teams should be encouraged to construct Building Stories which communicate the project's design outcomes and justifications in a cohesive manner. Building Stories support the dissemination of timely and relevant information throughout the project by deconstructing the project's complexities, relaying design justifications, encouraging participation and preserving the design process. Creating an environment where the Building Story can flourish is a challenge, but if achieved the team's collaboration effectiveness stands to be improved immensely.

2.1. Collaboration - the lifeblood of an architectural project

Effective collaboration is required if a project team is to conceive and construct architecture that can meet the identified requirements, budget and timeframe. Architectural design and construction is a complex problem and requires the input of many professionals working together in a variety of ways. This process is commonly referred to as architectural collaboration, and it plays an instrumental role in the success of the project. Unfortunately, there are a number of industry factors that are detrimental to effective collaboration, which if not overcome can impact the success of the project as a whole.

2.1.1. What is effective architectural collaboration?

Undertaking the design and construction of architecture is a complex, multi-faceted problem that requires input from a variety of professions. Given the interconnected nature of architectural requirements and decisions, project team members must regularly interact in order to exchange design decisions, discuss requirements and identify solutions to problems which may affect multiple parties. The term collaboration is commonly used to describe this activity, but ‘collaboration’ is an umbrella term, which can be differentiated in the following ways: (As found in: Mattessich & Monsey, 1992/1992, p. 39)

- *Cooperation* is characterised by informal relationships that exist without a commonly defined mission, structure or effort. Information is shared as needed and authority is retained by each organisation so there is virtually no risk. Resources are separate as are rewards.
- *Coordination* is characterised by more formal relationships and understanding of compatible missions. Some planning and division of roles are required, and communication channels are established. Authority still rests with the individual organisation, but there is some increased risk to all participants. Resources are available to participants and rewards are mutually acknowledged.
- *Collaboration* connotes a more durable and pervasive relationship. Collaborations being previously separated organisations into a new structure with full commitment to a common mission. Such relationships require comprehensive planning and well defined communication channels operating on many levels. Authority is determined by the collaborative structure. Risk is much greater because each member of the collaboration contributes its own resources and reputation. Resources are pooled or jointly secured, and the products are shared.

In the hypothetical, perfect architecture project, *collaboration* in its purest form is the ideal because it promotes an environment where the goals and actions of team members are perfectly aligned. This alignment, and its associated communication streams, would ensure the project’s design requirements and issues were identified and appropriately resolved in the shortest period of time. However, in the real world, such an environment is impractical because the industry is segmented, project budgets are limited, and time is in short supply. In this context, *collaboration* can generally only be achieved fleetingly because it is “time consuming and requires relationship building and is suited to very particular problems that

require such close coupling of the design process and its participants” (Kvan, 2000, p.413). Given this situation, striving for a pure *collaboration* environment within an architectural project team is ultimately impractical and uneconomical. Instead, effective architectural collaboration requires a more pragmatic approach, that appropriately applies *cooperation*, *coordination* and *collaboration* to move the project forward and resolve design issues.

Figure 2.1: Diagram illustrating the hypothetical benefit of cooperation, coordination and collaboration to architectural decision making and the effort required to achieve

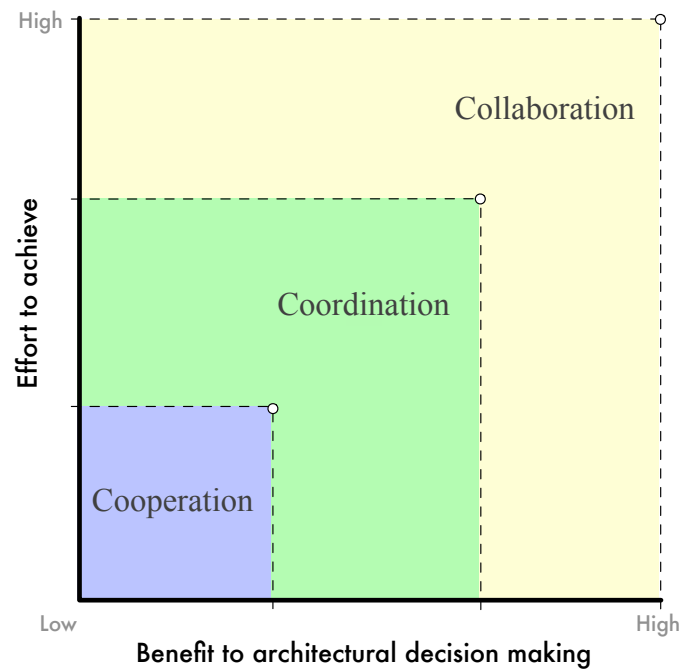


Figure 2.1 illustrates the theoretical effort required to achieve the three different forms of architectural collaboration, and the implied benefit to the decision making process. For the majority work undertaken by a project team, *cooperation* and *coordination* between participants is more efficient. These approaches suffice because from the perspective of individuals within the team, many design requirements are irrelevant, or do not significantly affect the work they are tasked with undertaking. However, certain design requirements and issues will warrant close *collaboration* between two or more members of the team. This activity can be justified because the collective value of these design decisions will outweigh the resource and time cost associated with the *collaboration* act. The challenge of a successful architectural collaboration strategy is knowing when and how to transition part, or all, of the team to a new working relationship. If these transitions do not take place, or take too long to occur, the benefit of these different architectural collaboration approaches may not be capitalised upon.

2.1.2. Barriers to effective architectural collaboration

A successful collaborative project must “establish a definition of the team, identify the desired outcomes, ensure there is a purpose in the collaboration and clarify the interdependencies of the members” (Arjun & Plume, 2006, p. 265). Establishing this shared understanding is an ongoing problem within the industry due to the complexity of most architecture projects, the segmentation of the industry and the limited timeframe in which most project-related activities occur (Emmitt & Gorse, 2003, p. 6). Dysfunctional collaboration and its detrimental effect on the success of architectural projects was highlighted in the influential 1994 UK government report ‘Constructing the Team’ (Latham, 1994). An underlying theme of the report being that more effective collaboration would bring “significant benefits by improving quality and timeliness of completion whilst reducing costs” (1994, p. 62).

Fifteen years on from ‘Constructing the Team’, the industry as a whole is still heavily segmented, and Latham’s vision of “partnering”, or Integrated Project Delivery (IPD) as it commonly known today (IIA, 2007), has seen only limited adoption. Achieving these cohesive environments within a project team is difficult because it requires the bridging of motivations, distance, skills and shifts in context. For example an office-bound architect is primarily concerned with describing their hypothetical solution, whereas the contractor on site needs to decipher the proposal and respond with perceived construction challenges. For the project to move forward efficiently, both parties must produce and request information at appropriate times which the recipient can understand. Any failure to identify misconceptions, mistakes, or knowledge gaps during this time can have serious implications for the task at hand, or impact other aspects of the project. These collaboration demands, coupled with financial and time pressures, has fostered a “strongly ingrained adversarial culture” (Egan, 1998, p. 9) within the industry.

Since the publication of the Latham report, the rapid and widespread adoption of Internet-connected computing devices has radically reshaped project communication by enabling instant, high-bandwidth communication between team members, irrespective of their geographic location. Considering the hypothetical potential of such functionality, one may have assumed that the collaboration effectiveness of the team would have improved at the same rate as these digital communication technologies were adopted. However, it can be argued that these digital innovations have served only to compound this collaboration challenge, because “anecdotal evidence suggests that we have started to focus too much on

the power and speed of systems, and not enough on the messages being transmitted, or the requirements of the users” (Emmitt & Gorse, 2003, p. 1).

2.2. Timely and relevant information benefits collaboration

The timely dissemination of relevant information can improve the effectiveness of collaboration within an architectural project by ensuring team members better understand, and enter into discussions about, design requirements and the justification behind design decisions. In so doing, the likelihood of confusion or conflict is reduced because participants are in a better position to monitor the development of the project, understand the consequences of the decisions being made, and the motivations of those behind them.

2.2.1. Understanding design requirements and decisions

In a perfect world, the requirements of an architecture project would be captured before the design process begins, and would remain static throughout its duration. To a degree this is what is attempted when a detailed client brief is prepared at the outset of a project.

However, no matter how thorough this document is, it can never embody all of a project’s design requirements for the following reasons:

- The client’s requirements change as the project evolves, primarily because their understanding of the built form improves as they work through problems with team members (Barrett, Hudson & Stanley, 1996, p. 2). The client’s ability to change the brief can be constrained, but this risks a built outcome that fails to meet their final requirements. Mitigating this risk “can only be achieved by sustained interaction with the client” (Barrett & Stanley, 1999, p. 35), which is often difficult due to time and financial constraints.
- Each participant brings their own set of design requirements and motivations that can play an influential role in the shaping of the architecture and the project timeline. As the majority of these requirements do not represent the explicit needs of the client, they generally go undocumented within the brief.
- Changes in the surrounding natural, financial or social environment can influence design requirements. For example, governments may introduce new legislation, or a financial crisis may force a restructuring of the project team, or its priorities.

The diverse range and dynamic nature of these design requirements, coupled with the segmentation of the project team, can lead to the “inadequate capture, structuring, prioritisation and implementation of client needs” (Evbuomwan & Anumba, 1998, p. 587). A means of overcoming this is to ensure participants can record, and be reminded of, the requirements which affect, or are affected by, the design decisions currently being made. For this to occur, participants need timely access to the requirements as they relate to the design issue being discussed. Alternatively, those responsible for including the requirement need to be informed of the impending decision, so that they can choose to be involved in the discussion, or review its outcome.

A project’s design requirements are deeply related to each other and it is likely that some design decisions will negatively impact project requirements, or existing aspects of the design. When this occurs “it is entirely the responsibility of the collaborators to ensure consistency, to consider all important alternatives, and to inform the partners of important justifications for decisions” (Lottaz, 2000, p. 1). To achieve this, those involved must efficiently identify the affected parties, understand the motives behind their requirements or decisions, and finally communicate a justified design alternative. Throughout this process, the delivery of timely and relevant information to the parties in question is critical because “poor co-ordination and communication of design information leads to design problems that cause design errors” (Emmitt & Gorse, 2003, p. 11).

2.2.2. Involving relevant parties in design discussion

As identified by Mattessich and Monsey (1992), effective collaboration requires the involvement of people with appropriate skills at appropriate times. Balancing these two parameters during an architecture project is a challenging proposition that can easily impact on the quality of the design, or the team’s ability to stay within budget. Ideally team members would only be involved in relevant design conversations, and at times when their input is most effective. Unfortunately, monitoring the majority of the issues and activities taking place is difficult because participants can easily be overwhelmed by the quantity of information, or worse, starved of meaningful contact due to professional or organisational segmentation. As described by McCall and Johnson (1997), this can lead to a situation where “team members too often do not know who they should be collaborating with at any given time... The central problem is that designers typically do not know when, with whom, and on what they should collaborate”. Communicating timely and relevant

information to all members of the team can mitigate this problem by fostering an awareness of the project's state and those influencing it. As long as the information communicated is both timely and relevant, its audience will be more able, and prepared, to digest it (Underwood, Maglio & Barrett, 1998).

An architecture project's scope, complexity and unknown variables increases the risks that are borne by its participants. These risks often bear rewards (McKenna, Wilczynski, & VanderSchee, 2006, p. 4), but it is generally accepted that where possible they should be avoided, or at least mitigated (Arrow, 2008). Increasing the team's awareness of the personalities and conversations taking place can reduce design risk by improving the project's decision making culture. This occurs because the behaviour of individuals often changes when they are aware that decisions, and their associated consequences, will be reviewed by relevant members of the team. For example it has been found that "individuals within groups would take greater risks, even if the consequences of the risk-taking would effect them personally. However, where group members were informed that failures associated with risk-taking would be openly disclosed to the group, there was a shift to less risky decision-making" (Emmitt & Gorse, 2003, p. 68). It is therefore important that a participant's design contributions are communicated to relevant members of the team in a timely manner. This will foster less risk within the team's decision making culture, resulting in improved collaboration effectiveness, and design outcomes that are more sensitive to the project's overall requirements.

2.3. Challenges communicating timely and relevant information

There are many barriers that block the distribution of timely and relevant information within an architectural project team. Unlike the majority of design and manufacturing endeavours, architectural projects present unique, multi-faceted problems, that require the intense involvement of many professional groups over long periods of time. Further complicating this process, project teams are temporary in nature, and their constituent parties are separated by knowledge, distance and professional boundaries that can impede and confuse the communication of collaboration information.

2.3.1. Architecture projects are unique and complex environments

An architectural project is characterised by the forces and requirements that shape its brief, influence design responses and affect the eventual built outcome. Martin, Heylighen and Humberto have identified six dimensions to a project which can influence this process (as found in: Martin 2003):

- **Actors** - individuals or groups of individuals who make decisions about a project based on their specific values (clients/owners, consultants, contractors, legislators, local community, ...)
- **Context** - the physical setting in which the project is built, including climate conditions, region, geological constraints, site boundaries, accessibility, transportation, ...
- **Organisation** - predetermined organisational structures that affect the outcome of a project (e.g. owner- builder delivery process, studio office structures, political control agencies, ...)
- **Practices** - operating procedures, use of tools, methodologies, precedent experiences, ...
- **Program** - user/client needs and requirements the project must accommodate within the given scope, time and budget
- **Resources** - the time and budget within which the project should be realised, as well as any types of documents, tools, conditions that provide a firm with special capacities to do so.

Against this complex backdrop exists the romantic notion that a single person is capable of overseeing all of these issues and determining appropriate design responses. However, “while most of the heroic mythology of design centres around the solitary designer, much of the most important work of architectural practice is in fact accomplished through skilful collaboration among the members of design teams” (McCall & Johnson, 1997). For these issues to be effectively addressed by appropriate team members, relevant information needs to be disseminated in a timely manner (Emmitt & Gorse, 2003, p. 120).

Communicating timely and relevant information to appropriate members of the team requires an extensive understanding of the project, its requirements and the roles participants have adopted. With this information at hand, participants are better equipped

to identify the communication pathways which link members of the team with relevant parties and project issues. As every architecture project is unique, participants commonly struggle with the scope of the problem, its issues, and their relationships with other team members. Identifying what constitutes timely and relevant information at the beginning of this process is difficult because the value and relationship system used to determine these properties has yet to form. Kruijff and Donath describe this as being where, “different people have different conceptual models (abstractions or interpretations) of the base model, which need to be connected in a coherent way to allow useful communication streams between the participants in the design process” (2001, p. 144).

The more participants interact with each other and the project’s design requirements, the clearer their understanding becomes of what constitutes timely and relevant information. For example, if services engineer has a clear understanding of the client and architect’s intentions for a space at the development stage, they will be in a better position to contribute to design decisions within this critical time period. This understanding cannot easily be transferred to another project because differences in design requirements, decisions and team composition may require the generation of a new value and relationship system. If this is the case, participants will need to shift the focus of what is communicated and monitored if relevant information is to be contributed and received in a timely manner.

2.3.2. Participants do not understand the justification behind decisions

Participants within an architecture project are more likely to communicate timely and relevant information when they understand the justification behind design issues and decisions. This is because:

- they have a better understanding of a design issue’s context, and are more prepared to make relevant contributions to discussion about it;
- they have a better understanding of the project’s requirements and motivations, and when isolated can make more informed design decisions;
- they can review the contributions of others in a more timely manner, and provide relevant feedback when appropriate.

Developing this understanding within a team is currently challenging because the industry emphasises the communication of design outcomes ahead of design justification (Evbuomwan & Anumba, 1998, p. 587). This primarily stems from the segmentation of the

industry, but it is magnified by a project's time and financial constraints. The prevailing emphasis on design outcomes does not mean most decisions are unjustified. However, due to the focus on outcomes, few resources are committed to recording or communicating the justifications behind decisions made.

The communication of standalone design outcomes would not pose a significant problem, assuming the underlying justification behind them can be derived easily. Unfortunately, the associated requirements and motivations behind a design outcome are often obfuscated because "design is a process that characteristically yields solutions that do not map neatly onto problems. Relationships between solution features and problem elements are often unpredictable in good design" (Cooper, Cerulli, Peng & Rezgui, 2005, p. 126). Therefore, even if a highly detailed description of the design outcome exists, team members may be oblivious as to how a project's underlying requirements were addressed. Without this understanding participants will struggle to critically review design outcomes, or reapply the lessons learnt during their development. This in turn can lead to unforeseen problems, repeated mistakes and the inability to identify issues or parties affected by design changes.

2.3.3. Project teams exist within a segmented industry

The AEC industry is notorious for its segmentation and conservative operating practices. For example in the United States there are approximately 700,000 AEC-related organisations (Becerik, 2004, p. 233), whilst in the United Kingdom the number is approximately 163,000 (Egan, 1998, p. 8). This splintered workforce, combined with the diverse skill-set needed to create architecture, means that most teams are comprised of multiple organisations, each of whom range in size, interests, capability and quality. Additionally, the manner by which many team members are selected for the project can severely restrict the flow of timely and relevant information. According to Egan, "too many clients are indiscriminating and still equate price with cost, selecting designers and constructors almost exclusively on the basis of tendered price" (1998, p. 8). This 'price first' approach often fails to take into account an organisation's ability to work with others, or any previous associations they have had with other team members.

The patchwork nature of a project team often leads to the architectural design process itself becoming fragmented. As this occurs organisations begin to work in isolation and design decisions are supplied to other parties as discrete parcels of information (see Figure 2.2). Often referred to as 'over the wall' syndrome (Egbuomwan & Anumba, 1998, p. 588), this

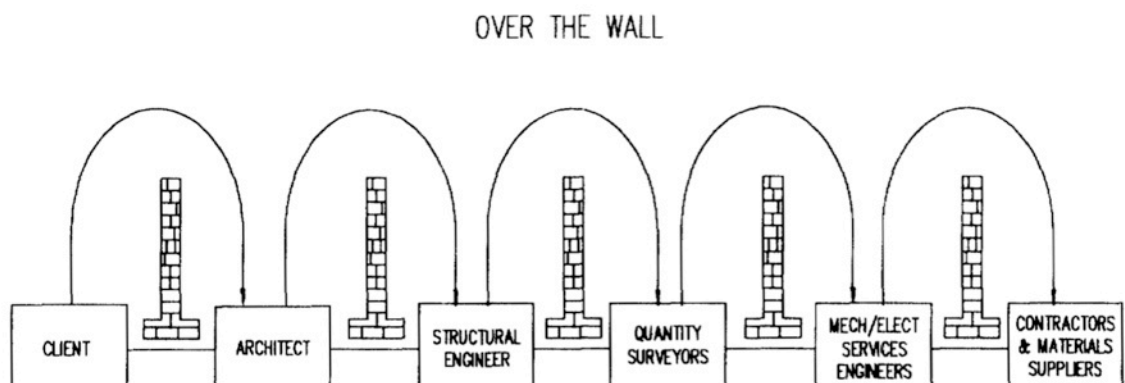
communication pattern bears a strong resemblance to the ‘waterfall model’ of software development (Leffingwell & Widrig, 2003, p. 24). Like ‘over the wall’ syndrome, the waterfall model for software development is also criticised for knowledge loss, time delays and the negative effects this has on the quality of decision making and the overall software product (Hibbs, Jewett & Sullivan, 2009, pp. 18-19).

The fragmentation of the design process leads to information barriers that can exclude third-parties from monitoring or participating in relevant design discussions. For those on the wrong side of the firewall their access to timely and relevant information can be impaired. Furthermore, these barriers impede the team’s ability to better understand the justification and relevant issues associated with design decisions that have made it ‘over the wall’. Evbuomwan and Anumba (1998) have identified the negative consequences of these impediments, which include:

1. elimination of viable design alternatives due to pressure of time;
2. prevalence of costly engineering changes and design iterations;
3. the lack of communication between each of the disciplines involved in the development process;
4. characterisation of the design process with a rigid sequence of activities;
5. constructability and supportability issues are considered late in the process;
6. fragmentation of the design data and difficulty in maintaining data consistency;
7. loss of information about design intent;
8. inappropriate estimation of product costs.

Figure 2.2: The ‘over the wall’ syndrome

(As found in: Evbuomwan & Anumba, 1998, p. 588)

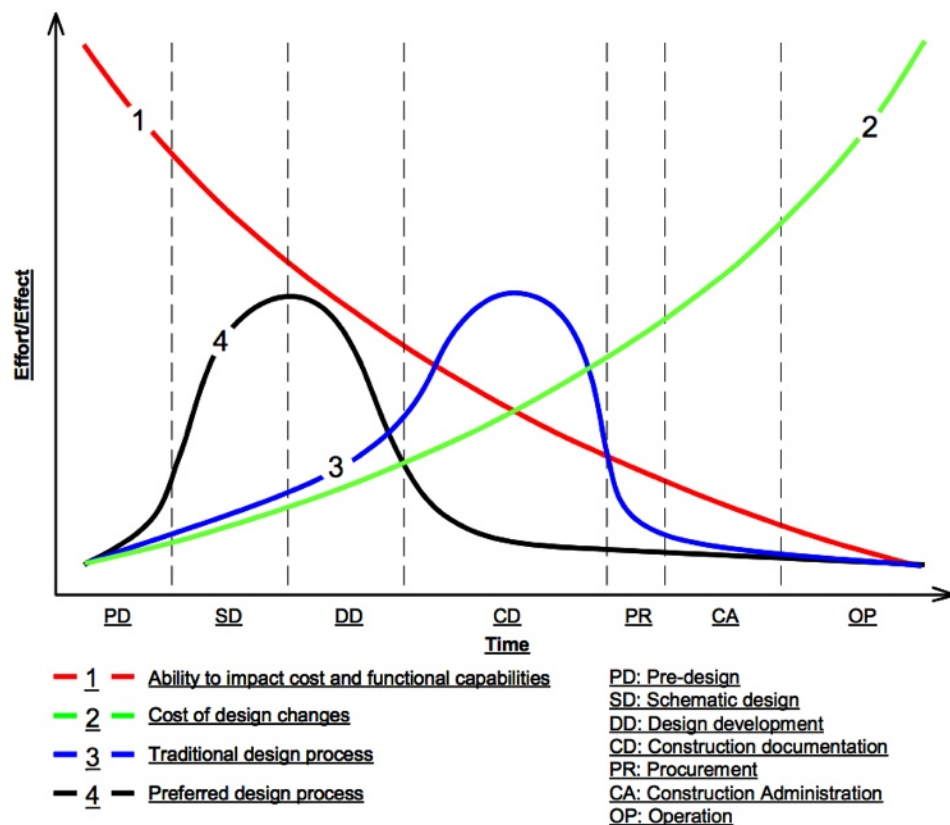


2.3.4. Time extremes restrict and erode project knowledge

Architectural project teams generally form quickly and are expected to enter into intense periods of collaboration soon afterwards. As stated by the Construction Users Roundtable, teams that are able to collaborate effectively in the early stages of the project “are most likely to achieve the desired outcomes: fast, efficient, effective, and cost-bound buildings” (2004, p. 4). Distributing timely and relevant information during the critical early stages of a project when many important requirements and decisions are being made is often difficult, because communication channels and understanding between participants have yet to form. As illustrated in Figure 2.3, the longer it takes for appropriate requirements and issues to be identified, the costlier it is to implement design changes. If timely and relevant information is available from the earliest stages of the project, then the environment is more conducive to collaboration and the early identification and resolution of design requirements and issues.

Figure 2.3: A diagram showing the relationship between effort and effect during the architectural design and construction process

(As found in: The Construction Users Roundtable, 2004, p. 4)



Depending on project requirements, the time needed to design and construct the architecture can range from many months to years. During this time, the composition of the team will change as the project evolves and those involved leave or adopt other roles. However, “even when the original participants are no longer involved, their decisions and actions still impact the project” (Kalay, 2001, p. 742). This is an impediment to the dissemination of timely and relevant information because some, perhaps even all, of the original contributors of the underlying data may no longer be involved in the project. Additionally, even if the contributor is still involved, this does not necessarily mean they can reliably provide, or recollect, relevant information associated with the design requirement or decision. When these knowledge gaps form, “there is considerable potential for misunderstandings, inappropriate changes, which give rise to unforeseen difficulties, decisions which are not notified to all interested parties” (Peng *et al.*, 2000, p. 290).

For many years digital collaboration tools have been seen as a promising means of overcoming this knowledge retention problem because they provide a reliable means of storing large quantities of data indefinitely. However, it is not simply a question of capturing data, instead for knowledge management “to be successful, it must be understood, disseminated, fostered, expected and compensated; implemented, aligned with business objectives, measured, and constantly renewed” (Pulsifer, 2008). This is a difficult undertaking for a segmented project team, hence it is unsurprising that digital knowledge management is uncommon within them, or many of their constituent organisations.

2.4. Improving collaboration through Building Stories

There are many aspects of an architectural project which impede the dissemination of timely and relevant information throughout the team. A viable means of overcoming these problems is to promote the creation and propagation of the project’s Building Story (Martin *et al.*, 2003). The Building Story describes the evolution of the project from the perspective of those involved. This is achieved by encouraging participants to reflect on, describe and relate the activities and artefacts that are associated with the collaborative design process. By translating the design process into a series of interconnected events, the ability for the team to understand the requirements, issues and state of the project is collectively improved. This retrospective process, and the tacit knowledge which is gained during it, improves the collaboration effectiveness of the team by enabling participants to better communicate and identify timely and relevant information.

2.4.1. The journey is as important as the destination

Ensuring the timely distribution of relevant information within the project team requires insight into the unfolding design process, the issues affecting it, and the personalities involved. To gain this perspective, participants need to figuratively, and at times literally, read the stories taking place inside the project. Architecture stems from the client's initial requirements, and continues to evolve until the built form ceases to exist. Martin, Heylighen and Cavallin (2003) have termed this ongoing narrative the project's Building Story, and like any story, its purpose is to clearly articulate a series of events involving various characters to an audience. In a Building Story, the characters are the members of the team, and the events are the decisions, actions and artefacts they generate during the design process. A Building Story's audience changes as time passes. Initially it will be the project team itself, but as it disbands this role will be occupied by those who maintain, redevelop and reflect on the history of the architecture. Martin, Heylighen and Cavallin (2005) have demonstrated that the process of recording and experiencing Building Stories supports the exchange and retention of tacit knowledge, which "expresses itself in human actions in the form of evaluations, attitudes, points of view, commitments, motivation, etc." (2007, p. 66). By experiencing this story and absorbing its tacit knowledge, a team member's ability to identify and resolve relevant issues in a timely manner is improved. This is made possible because "the story format provides a dense, compact way to deal with and communicate the complex reality of a real-world project, while respecting the interrelated nature of events, people and circumstances that shape its conception" (2005).

Using stories to convey complex information to a broad audience is as old as speech itself. The rise of the information economy and knowledge management has lead many to reconsider the role and value of storytelling in business (Denning, 2004). Martin *et al.*, argue that while other industries have embraced storytelling as a viable communication and teaching tool, formal application within the AEC industry has, for the most part, been limited to the field of research (2005, p. 35). A considerable amount of this research has centred around the development of Case-Based Reasoning (CBR) tools which are intended to help solve architecture problems (Domeshek & Kolodner, 1993). CBR is related to the field of artificial intelligence, and at its core is the process of solving new problems by applying lessons learnt from past experiences. The recording and studying of stories is an intrinsic part of CBR, but unlike Building Stories, this process is applied to historical data rather than the project at hand. CBR is valuable to architectural collaboration because it

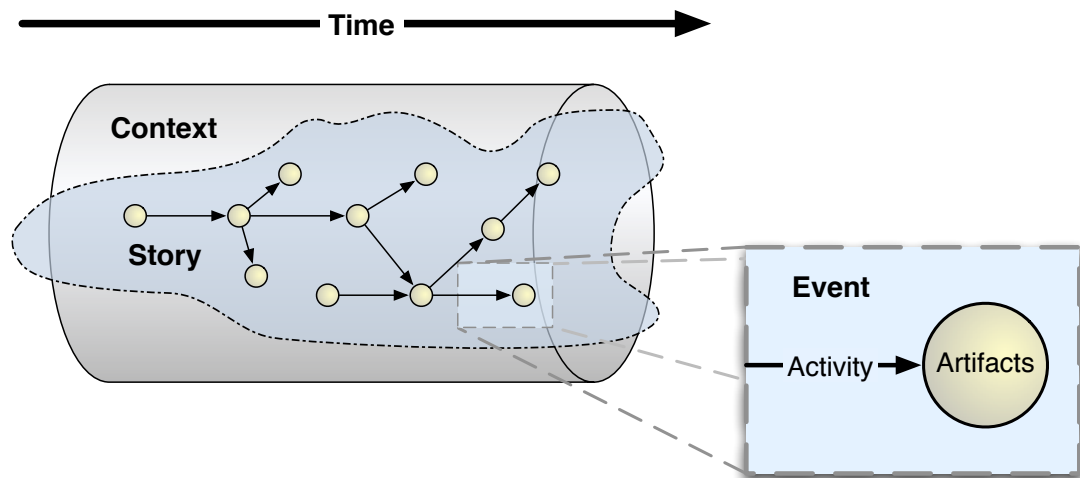
allows teams to draw on the collected experiences of a broad range of projects and people, and in so doing allows potential issues and solutions to be identified in a timely manner. Beyond helping to identify solutions to current design problems, CBR systems “that make evaluative information available to designers early in design, can contribute to designers' awareness of the downstream implications of their decisions” (1993, p. 90). Unfortunately CBR’s emphasis on historical information means it is not suited to recording or broadcasting the design process currently taking place. Consequently, whilst CBR may identify a relevant historical precedent for a given problem, it does not present participants with the sequence of events that led to it, or shed light on the actions taking place elsewhere that may influence its resolution. An architecture project’s idiosyncrasies and complexity makes identifying relevant historical cases is a difficult process (Schneider & Petzold, 2009, p. 206). When this fails, team members risk being presented with irrelevant information, which in turn can impede the effectiveness of collaboration.

In contrast to the retrospective process of CBR, a Building Story records and distributes the ongoing design process with the explicit purpose of increasing the awareness and understanding of those involved. As illustrated in the work by Martin, Heylighen and Cavallin (see Figure 2.4), the project’s own timeline and context form an intrinsic part of its Building Story.

“Within this time and setting, the story itself can be conceived as a network of events, which altogether form the path from start to end. Each event is made up of two types of building blocks: activities and artefacts. Activities embody (an account of) the actions and interactions performed by the actors during the event, and link to artefacts they create or use in doing so” (2005, p. 38).

Activities range from brainstorming design concepts through to onsite construction meetings. Artefacts may be as simple as a hand-drawn sketch, or as complex as a fully developed digital model. Unlike CBR, the events which link these two elements are formed by the team members themselves for the purpose of describing and understanding the process unfolding around them. In this sense, events are the conceptual glue which bonds design outcomes (artefacts) to their respective development paths and underlying justifications (activities).

Figure 2.4: An illustration of a Building Story's components and their relationship to each other (Based on diagrams in: Martin *et al.*, 2005)



The cornerstone of a Building Story is its events, which are explicitly defined by appropriate members of the team. As no single person is involved in every facet of the project, constructing an accurate Building Story requires contributions from the entire team. In so doing, participants are encouraged to review, record and communicate the actions they took and the artefacts they generated in the process of reaching an acceptable design outcome. Building on the work by Domeshek and Kolodner (1993), Martin, Heylighen and Cavallin have identified four different story types that unfold during the creation and retelling of these events (As in: Martin *et al.*, 2003, p. 4):

- **Point stories** - describe how certain features of a design (e.g. separated entrances) contribute towards, or undermine some particular goal.
- **Interaction stories** - discuss how features of a design case can be interpreted with respect to several design goals (e.g. privacy, security, circulation), perhaps advancing some while frustrating others.
- **Cluster stories** - serves mainly as a table of contents by summarising several point stories that are located close to one another (e.g. all stories about a particular room).
- **Design stories** - annotated text, collected by analysing critical writings, describes a conceptual point that characterises the uniqueness of the design.

A single event can be comprised of any number or combination of story types. The activities referenced describe how these stories came about, whilst the resulting artefacts demonstrate the influence the event had on the project. From the perspective of the overall

Building Story, these story types provide a high-level index by which participants can navigate between, and better understand an event's component activities and artefacts. When portrayed against the backdrop of the project and its timeline, these interconnected events help to expose and preserve the project's history. This overall process is conducive to effective collaboration because it encourages participation and retrospection, two important behaviours when promoting the timely distribution of relevant information.

2.4.2. The benefit of Building Stories to the collaborative design process

The act of constructing and sharing a Building Story benefits the collaborative design process because it reinforces the idea of design as a circular process. For the purposes of simplicity, architectural design is often thought of as a linear process that flows smoothly from requirements gathering through to conceptual development, documentation and construction. This illusion of a linearity frequently manifests itself in the structure of the team and communications patterns such as 'over the wall' syndrome (see Section 2.3.3). Blindly handing off design outcomes to other team members places a strain on the design process because it is difficult to understand the design's intentions, justifications, or evaluate its success and influence relative to the overall project. Difficulties arise because design is not straightforward, instead the endpoint is constantly influenced by the events which transpire during its conception. To compensate for these shifts, the collaboration system used should support Zeisel's notion (1984) of design as a circular process comprising of three interconnected actions:

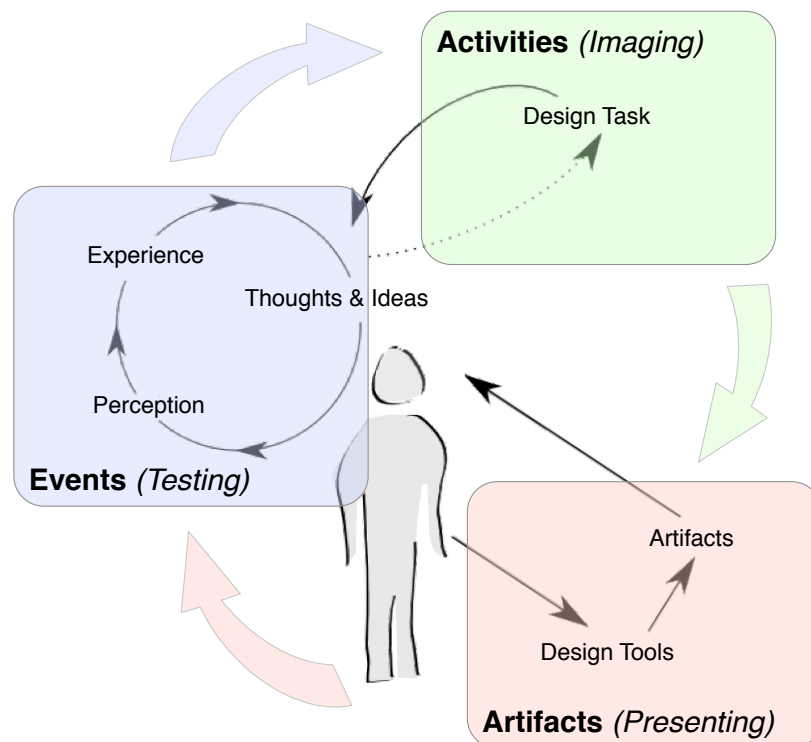
- **Imaging** - "means forming a general, sometimes only fuzzy, mental picture of a part of the world" (1984, p. 8). It is the act of conceiving a solution to a design problem, which can involve debating potential solutions with team members through to sketching potential concepts.
- **Presenting** - "includes the very important characteristic that for each design one must choose and organise only some elements from a larger number" (1984, p. 8). It is the development and communication of the ideas established during the imaging process, through the use of visual aids such as drawings and models.
- **Testing** - "is a feed-back and feed-forward process, adjusting the relation between a design product as it develops and the many criteria and qualities the product is intended to meet" (1981/1984, p. 9). It is the critical review of the presented idea

based on its ability to meet project requirements, design goals and its performance relative to other design alternatives.

In an architectural collaboration situation, a design conversation may lead to a series of sketches (*imagining*). These ideas are further developed using a 3D digital model, which are distributed throughout the team as a series of 2D plans and renderings (*presenting*). Relevant team members then evaluate the design to decide whether it meets the project requirements (*testing*). Conclusions from this review, such as finalised architectural elements, altered project requirements and new design issues, are fed into the next imaging process. This circular pattern “means that there is no clear segregation between imaging, presenting and testing, but a significant relationship in which each depends on the other” (Schneider & Petzold, 2009, p. 207). Communication patterns which break these relationships harm the design process. The Building Story’s use of artefacts (*imagining*), activities (*presenting*) and events (*testing*) reinforces this circular pattern (see Figure 2.5), which leads to a more consistent, and therefore effective, collaborative design process.

Figure 2.5: Design as a circular process and its relationship to the events, activities and artefacts within a Building Story

(Based on diagram in: Schneider & Petzold, 2009, p. 206)



2.4.3. Using Building Stories to improve timely access to relevant information

Building Stories improve the team's ability to access timely and relevant information because they deconstruct the project's complexities, expose its design justifications, encourage participation and preserve its history. These factors work in unison to overcome the barriers which restrict the project team's ability to access timely and relevant information identified in Section 2.3, namely:

- **Architecture projects are unique and complex environments** - Building Stories deconstruct project complexity by conveying the core ideas and processes behind an architecture project. To achieve this, those involved retrospectively organise and relate the project's diverse range of activities and artefacts into a series of interconnected events. These events, and the stories that comprise them, break down and describe the project's core requirements, issues, outcomes and hierarchical structure. Through the creation of an overriding project narrative, participants are provided a reference point through which they can better understand the dynamics of the project and their role within it.
- **Participants do not understand the justification behind decisions** - Building Stories expose outcomes by outlining the relationship between a project's activities and its artefacts. With this knowledge, participants are able to discern how and why a design outcome was conceived, and what processes influenced its development. This is achieved through the telling of stories which are "about problems, and how and why they got—or, more likely, didn't get—resolved. They include a description of the problem, the setting, and the solution... Though unashamedly unentertaining and lacking most elements of a conventional story, they are nonetheless the uncelebrated work-horse of organisational narrative" (Denning, 2004, p. 127).
- **Project teams exist within a segmented industry** - Building Stories encourage participation because the recording and organising of activities, artefacts and events can only occur if team members are free to interact with each other. If one or more participants are excluded from this process, then the integrity of the Building Story is diminished and there is a risk that the circular design process may be broken. Assuming all members of the team can contribute, the Building Story grows organically as the various contributions are assembled. This process will generally begin with "a story told by one member of the group. Ideally, that first story sparks

another, which sparks another. If the process continues, group members develop a shared perspective, one that creates a sense of community” (Denning, 2004, p. 126).

- **Time extremes restrict and erode project knowledge** - Building Stories preserve and expose a project’s design process because they encourage team members to periodically describe the work they have undertaken and relate it to the broader project. This process helps to maintain a history of the design decisions so that third-parties are able to construct a more complete understanding of the project and its evolution. Third-parties can range from current team members, through to people associated with the architecture after the design team has disbanded. The events, activities and artefacts described within the Building Story provide an insight into the unfolding situation, the project’s requirements and the justification behind the design outcomes that have been collectively agreed upon.

2.5. Summary

Disseminating timely and relevant information is instrumental when trying to create an effective collaboration environment within an architecture project. If this information is available, the team’s collaborative decision making process is improved because collectively its members are more aware of the project’s requirements, issues and potential consequences of their actions. Unfortunately, the complex and time sensitive nature of an architecture project, coupled with the segmentation of the AEC industry, restricts, and at times blocks this flow of timely and relevant information within the team. These factors have a detrimental effect on the collaboration effectiveness of project teams and is frequently cited as a significant failing of the industry. Building Stories are a means of improving the collective understanding of design teams because they overcome the impediments to timely and relevant information. A Building Story achieves this by using the concept of events to link a project’s activities and artefacts into a cohesive narrative. As participants experience and contribute to this narrative, they gain a better understanding of the project’s current state, its requirements, and the significant issues faced by the team. With this knowledge, team members are more able to identify and communicate timely information and take part in relevant design conversations.

The communication and collaboration landscape of the AEC industry was radically altered during the closing decades of the 20th Century, with the rapid and widespread adoption of desktop computing and the Internet. Today, the majority of the project team’s

communication and collaboration interactions occur digitally. In general, the AEC industry is still struggling to come to terms with the collaboration possibilities and challenges of this new digital environment. For example, it is now possible for any member of the project team to immediately communicate large quantities of digital information to almost any location. Although this has enabled more frequent interactions between team members, the vast increase in project information and technical complexity that has come about as a result poses new problems. In an effort to address some of these issues, a current industry trend is the consolidation of project information within centralised and highly structured Building Information Models. This technology generates management and process benefits, but its complexity restricts the ability of many within the team to contribute to the design process. The next chapter reviews this digital migration, the rise of the Building Information Model, and their combined influence on the team's ability to access timely and relevant information.

3. The Digitally Fragmented Project Team

Architectural collaboration in the age of the Internet and Building Information Model

The tools the Architecture, Engineering and Construction (AEC) industry use to collaborate are a deciding factor in whether team members can access timely and relevant information about the project and its design conversations. The introduction of Internet-based digital collaboration technologies have radically changed the project team's toolset and communication processes. This new generation of tools have eliminated many long-standing collaboration barriers, and provided a new range of interaction possibilities. Unfortunately, the uneven adoption of these highly capable communication tools has introduced new collaboration barriers, and magnified existing information management problems. To meet these challenges, the AEC industry is turning to the Building Information Model (BIM) to more efficiently capture, process, and present digital information within a project team. Unlike the relatively simple digital models which preceded it, a BIM is a comprehensive, virtual representation of the proposed architectural design. Although this helps form an excellent understanding of the architecture, the complexity and centralised nature of BIM means the majority of the team cannot interact with it. This participation barrier further inflames the digital fragmentation within the team, and complicates access to timely and relevant information.

3.1. Architectural collaboration's digital migration

In a short period of time the Internet has reshaped the tools and processes the AEC industry uses to collaborate. This migration to a digital collaboration environment allows project teams to exchange more information, more frequently, and over greater distances.

Unfortunately, these changes have not come without a price. Project teams are struggling to come to terms with the quantity and density of the digital information generated, and irregular adoption of technology has created significant disparities between members.

When combined, these impediments pose significant challenges that impair the effectiveness of digital collaboration within the team.

3.1.1. The elimination of distance and bandwidth restrictions

The introduction of low-cost, high-bandwidth Internet connections radically reshaped the communication and collaboration patterns of the AEC industry. Prior to the mainstream availability of Internet connectivity during the 1990s, the AEC industry had relied on collaboration methods which had gone relatively unchanged for centuries (Wilkinson, 2005, p. 20):

- **Printed documentation** - The exchange of printed architectural drawings and documentation was the primary means of communicating design intent. Fax machines allowed this information to be instantly relayed, but due to the loss of print quality, the majority of documents were exchanged directly, by post, or courier.
- **Physical models** - Scale models of the design and its construction details were used to demonstrate the spatial characteristics of a proposed solution. Given the cost and time to prepare a physical model, they were generally used sparingly and in support of printed documentation.
- **Sketches and photographs** - Sketching was used as an efficient means of quickly communicating an architectural concept or solution to a design problem. Effective collaboration using this media generally required close proximity as significant lag in the feedback loop would stifle design conversations.
- **One to one conversations** - Informal discussion between team members either in person or via the telephone was the most prevalent means of collaboration within a project. With these modes of communication, unless those involved were conscientious, there was often no record kept of what was discussed, or the conclusions drawn.
- **Group meetings** - Formal meetings that involved multiple parties were key to assessing a project's current status. If people were not able to physically attend a meeting the option to join via telephone was often available, but this was not ideal as group meetings are more effective when participants are able to see those involved (Isaacs & Tang, 1994). Records of outcomes of these meetings were distributed throughout the team, either in person, or via fax, mail or courier.

Although these methods satisfied the team's collaboration requirements, their reliance on physical information exchange and proximity imposed many limitations:

- **Quantity** - Information presented in printed documentation can be dense, but it is ultimately limited by the media's physical properties. To compensate, the industry has established a series of drawing type, scale and annotation conventions. However, even with these in place, a set of printed drawings still struggles to convey a project's many details, requirements and issues.
- **Speed** - Traditionally the speed of architectural collaboration was inconsistent at best. Telephone conversations are instantaneous, but the scope and quantity of information communicated is limited by the time available to both parties. In contrast, printed documentation succinctly conveys a great deal of information, but its distribution is time consuming. Fax machines can relay documentation instantaneously, but the associated loss of print quality is problematic given the significance of scale and fidelity in most architectural documents.
- **Cost** - Distributing printed documentation, constructing physical models and organising multi-party meetings has associated costs that are relatively high. Conversations in person or over the telephone are financially efficient, but the disruption they cause impacts the productivity of both parties (Mark, Gudith & Klocke, 2008, p. 107).
- **Frequency** - The frequency of most collaboration interactions was traditionally restricted by their associated cost and time delays. Delivery of printed documentation is often dictated by delivery schedules, whilst meetings depend on the availability of relevant team members. The introduction of cellular phones has removed the time and location restrictions on one to one conversation (Beyh & Kagioglou, 2004), but exchanges are limited to relatively simple voice or text based information.

The ability to quickly pass large quantities of digital information between a broad range of Internet-connected devices negated many of the above shortcomings associated with traditional architectural communication. This connectivity has transformed the industry's perception of computers from standalone tools, into essential tools for collaborating and interfacing with the project's knowledge base.

3.1.2. Immediate and frictionless collaboration interactions

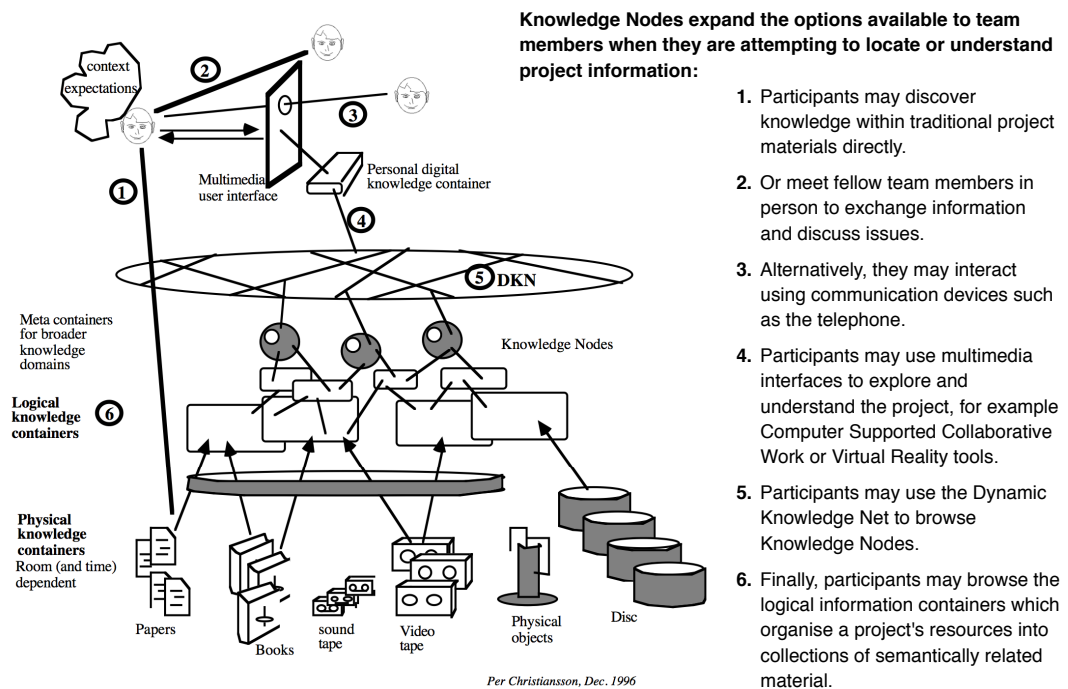
The bottleneck of traditional architectural collaboration was the time and effort required to physically pass information between team members. The availability of high-speed, low-cost Internet connectivity led to a significant and growing increase in the quantity and frequency of information transfer within the project team (Howard & Petersen, 2001, p. 8).

3.1.2.i. Boundless information transfer

Internet-based tools allow team members to exchange large quantities of information and easily access the digital resources associated with a project. Team members working in a purely digital environment are not limited by physical constraints, and communication incurs minimal disruption or financial cost. With this freedom, participants do not need to place as much attention on editorialising information they wish to communicate. Rather than selecting a few key documents to physically convey, it is now easier to send a collaborator all the relevant digital files they are actively working on. From the perspective of the sender, this approach saves time and minimises the chances that the recipient will need to request more information, or make unwanted assumptions.

Hyperlinks embedded within the communicated data allow collaborators to easily and reliably reference digital resources that are available on the Internet or local network. This capability is unrivalled by traditional collaboration methods, and it means collaborators can explicitly define relationships between digital resources, rather than having the recipient do this manually. Beyond the simple linking of digital resources, research by Christiansson (1996) (1998) has explored how Semantic Web technologies (Lee, Hendler, Lassila & others, 2001) could be used to weave project information into a series of Knowledge Nodes (see Figure 3.1). These Knowledge Nodes would aggregate and present the project's disparate resources to participants in a manner that would allow them to better understand what information is available. For example, a Knowledge Node could be used to describe the project's architectural concepts and what resources define or exhibit them. Knowledge Nodes would enable participants to explore the project's knowledge base more intuitively. For example, when viewing the project's digital model, a Knowledge Node could highlight the relevant briefing documents that outlined design requirements. Although this concept, and its related technologies, have yet to gain momentum within the industry, it does serve to illustrate the largely untapped potential of the Internet for increase the richness and usefulness of the information presented to team members.

Figure 3.1: The use of Knowledge Nodes to support information discovery through the weaving together of project resources (As found in: Christiansson, 1998, p. 203)



In conjunction with a new generation of software tools, high-speed Internet connectivity is allowing distributed teams to simultaneously access and interact with shared digital resources. This centralisation removes the risk associated with data duplication and miscommunication, and benefits collaboration as design changes are reflected immediately across the entire team. Creating shared virtual spaces has been a recurring theme of architectural research (Chen & Maver, 1996) (Soubra, *et al.*, 2000) (Christiansson, *et al.*, 2002). As yet, practical and technical problems have limited the application of these environments within typical project teams. However, this may soon change thanks to the growing capability and influence of consumer-orientated virtual worlds such as Second Life and Google Earth. It has been shown that architectural project teams can successfully leverage these services (Angulo, Fillwalk & Vásquez de Velasco, 2009), and they are inspiring new generations of AEC-specific software (Harrison, 2008).

3.1.2.ii. The overcoming of geographic and organisational barriers

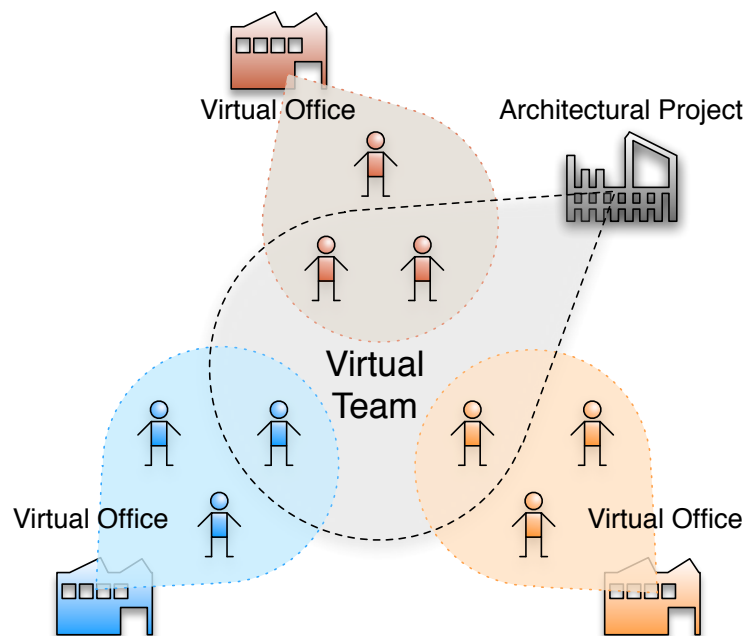
Internet-connected team members can efficiently, securely and reliably exchange digital information, regardless of their geographic distribution. As a result, effective architectural collaboration is no longer dependent on close proximity, delivery schedules, or fixed

locations where team members can meet and work productively. Two fundamental, Internet-centric working environments have emerged from these changes:

- **The virtual office** (Kern, 2008) provides participants with greater working flexibility as it allows them to access many of their employer's tools and resources over a secured Internet connection.
- **The virtual team** (Jamieson, Thorpe & Tyler, 1996) fosters relationships within a distributed team by providing a common, online space for conversation, community building, and the exchange of sensitive project information.

Figure 3.2 illustrates how these technologically similar, but conceptually different, digital environments relate to each other, the architectural project, the members of the team and their respective organisations.

Figure 3.2: The relationship between the virtual office and the virtual team



The virtual office exposes an organisation's tools, processes and resources on the Internet where they can be accessed by employees working from remote locations. This accessibility benefits architectural collaboration as team members have more opportunities to productively contribute to a project. A virtual office's functionality typically includes the ability to interact with an organisation's communication systems, project records, processes and accumulated knowledge. Although many industries have used virtual offices to reduce physical office space, within the AEC industry it has instead been used to facilitate the

creation of temporary, project-specific office spaces that are physically separate to that of the employer's. Allowing the entire project team to work in the same physical space improves the bonds between them and negates many potential collaboration barriers. Traditionally, setting up a dedicated, physical office space for such an activity would be difficult because it would lack many of the tools and resources of the employer's office. Thanks to virtual office technologies and wireless/mobile Internet connectivity this is no longer the case, and such spaces can be created relatively quickly and affordably.

A virtual team helps to facilitate and strengthen relationships between people from different organisations and geographic regions. Compared to a virtual office, the functional requirements of a virtual team differ significantly, even though the enabling technologies remain the same. These differences arise because a distributed project team lacks the trust and uniformity that an employer can impose on its employees. In comparison, virtual team members are generally unwilling to be completely open with collaborators, and so the technologies and processes must be relatively straightforward and generally accepted.

“A virtual team must operate with a technology and at an acceptance level that is available to all members. If one team member does not have, or will not use, a chosen technology, they will feel left out. In this circumstance you risk creating two teams, one of ‘haves’ and the other of ‘have nots’” (Falkowski & Troutman, 2005, p. 113).

When these trust, process and technology needs can be met, virtual teams have demonstrated an ability to overcome the geographic and organisational barriers which have traditionally restricted collaboration within project teams (Alexander, *et al.*, 1998, p. 5).

3.1.2.iii. *Virtually free communication*

Internet-based collaboration requires a significant technology investment, but compared to traditional methods of information exchange it is more cost-effective. A prominent area of savings is in the transfer of formal documentation between different participants or locations. Husin & Rafi identified that “the use of iCAD (Internet-enabled CAD) in the working drawing phase is able to reduce the time and cost of multiple drawing production, printing and delivery” (2003, p. 511). According to studies by AEC software vendors, savings on printed documentation can outweigh the cost of migrating teams to a digital collaboration environment (Asite, 2006) (Autodesk, 2007). Although, there is potential vested interest in these studies which undermines the credibility of the findings.

Email is an ubiquitous form of digital communication that delivers a wide range of savings compared to conventional alternatives. It is often preferred by team members as there is a perception it is less disruptive than the telephone because “the phone call interrupts and the email does not, as recipients deal with emails at their own convenience” (Davenport & Pearlson, 1998, p. 57). Additionally, email can be effortlessly delivered to multiple parties, and multiple file attachments can be included at no extra expense. Combined, these factors have significantly reduced the cost of project communication, which has increased the frequency of interactions between participants, and quantity of information exchanged.

3.1.2.iv. More frequent interactions within the team

Team members are presented with more opportunities to monitor and participate in project-related conversation thanks to the widespread availability of low-cost, high-speed Internet access and the broad array of Internet-connected devices. The flexibility this provides has increased the frequency of interaction between team members (Gorczyca, 2002, p. 379), which in turn benefits collaboration because “an increased amount of communication received by employees has positive effects on their organisational knowledge and commitment” (Edmunds & Morris, 2000, p. 21).

Research by Lam, Chua, *et al.* (2005) has shown that a virtual project team must mature before interaction frequency will increase and reach productive levels. As this occurs the team passes through a series of interaction phases which are described in Table 3.1.

Collaboration interactions can be enhanced at each phase of the maturing team if the appropriate digital collaboration tools are applied. For example, email is very useful during a team’s early phases, but as the team matures it can be perceived as being difficult to manage and lacking in immediacy and richness. At this stage other digital collaboration tools come to the fore, such as instant messaging, voice/video conferencing and project intranets. A failure to recognise these phases, or apply the appropriate tools, can have a detrimental effect on collaboration.

Table 3.1: The four-stage model of virtual team maturity

(As found in: Lam *et al.*, 2005, p. 359)

	Interaction	Organisation	Modus operandi
Chaotic	No or minimal interaction.	No defined roles, responsibilities or division of work.	Neglectful, where work is frequently incomplete and learners show little care and attention.
Surviving	Occasional work-related interaction, lack of deep discussion.	Loosely defined roles, responsibilities and division of work.	Reactionary, where work is completed in an ad hoc manner at the last minute.
Organised	Frequent work-related interaction, some social exchanges and some deep discussion.	Clearly defined roles, responsibilities and division of work.	Planned, where work is completed in a systematic manner but where the team may not cope with unexpected events.
Thriving	Frequent work-related and social interaction, much deep discussion.	Clearly defined roles, responsibilities and division of work, and willingness to help others.	Planned, where work is completed in a systematic manner and the team is able to deal effectively with unexpected events.

3.1.3. The digital downside of Internet-based architectural collaboration

The migration of the AEC industry's communication processes to the Internet has not come without a price. The abundance of data, coupled with the industry's haphazard adoption of technology, has created new problems which must be overcome if the team is to collaborate effectively.

3.1.3.i. Coping with the influx of digital information within a virtual vacuum

Alleviating the restrictions associated with traditional collaboration methods has highlighted information management problems and the fragility of the team dynamic. The density and quantity of digital information exchanged, coupled with the friction resulting from a lack of social presence within virtual teams, is hindering the ability of team members to participate in design conversations.

Information density - Traditionally the maximum amount of information exchanged was constrained by the physical properties of the medium used to convey it. For example, the detail in a drawing is limited to what people can easily see, whilst a physical model has a

fixed scale and purpose. Digital exchanges are not tied to the physical world, and as a result they can convey an almost limitless amount of information. Nowhere is this capability more evident than within the project's digital model. Initially translations of conventional drawings, digital models have evolved into comprehensive, 4-dimensional Building Information Models (BIM). Whereas "plans and sections, the traditional conventions of architectural communication... are a linguistic system, a visual, graphic language" (Ambrose, 2006, p. 183), in contrast "BIM offers the double-edged promise of displacing abstraction with simulation... The virtual building model is the thing as well as the representation of the thing. There is no abstraction" (2006, p. 184). This emphasis on simulation over representation provides many process benefits, but the associated complexity can impair collaboration. Conventional drawings are generally concise and carry an explicit meaning, which makes them easier to understand and place within the broader design conversation. Digital models lack this clarity because they depict an unfiltered view of the architecture in all of its detail. Additionally, the complexity of a digital model often makes direct manipulation by other parties difficult or impossible. Some consider this a benefit as it allows them to act as digital gatekeepers, but this limitation diminishes the team's capacity for collaboration.

Digital overload - The improved speed and frequency of Internet-based communication is only of benefit if team members can process it all (Edmunds & Morris, 2000, p. 21). Unfortunately, the large number of irrelevant messages communicated during the course of a project complicates this task. The minimal cost and effort associated with email is a contributing factor in this, as there is no tangible cost to sending poorly conceived and considered messages. Additionally, the perception that recipients are not disrupted by digital messages is a falsehood, because research has shown that email disrupts work patterns almost as much as telephone calls (Jackson, Dawson & Wilson, 2003, p. 57). If left unchecked, this barrage of digital messages leads to "lost information, and reduced responsiveness" (Whittaker & Sidner, 1996, p. 284), and an overall decrease in collaboration effectiveness.

Project collaboration is further complicated by the sheer number of digital artefacts that are created during the design process. Conventional printed documentation and physical models require an investment in time and resources to produce. In contrast, digital files can be created, copied, and exchanged effortlessly. This has enabled more frequent and efficient information exchange, but it has led to a significant increase in the number of

artefacts that must be managed and maintained. This change has highlighted deficiencies in the information management skills of many teams. For example, research undertaken by Gross *et al.* found that “a mountain of files were generated by participants from different universities. It was a difficult task determining which and how some of these files related to each other” (1998, p. 466).

Distance: Geographically distributed teams can effortlessly communicate over the Internet, but “managing problems in virtual teams is different from managing those that arise in other teams. It is difficult, for example, to develop the necessary trust to resolve issues when working in the virtual office” (Davenport & Pearlson, 1998, p. 60). When working in close physical proximity team members glean a lot of tacit information about each other and their processes. Deriving this understanding in a virtual environment is more difficult, because interactions between team members are “less frequent and more deliberate and intentional compared to a face-to-face context” (Garrison, 2006, p. 27). Without this tacit information, participants will lack “an understanding and respect for each other and their respective organisations: how they operate, their cultural norms and values, limitations and expectations” (Mattessich & Monsey, 1992/1992, p. 12). To address this shortcoming, Garrison recommends that virtual design teams promote a “climate that will create a community of inquiry”, where participants “have an opportunity to interact formally and informally with peers” (2006, p. 27). Unfortunately, creating this environment within a distributed architectural project team is difficult as the industry is segmented, and the majority of the digital tools currently used do not promote such interactions. For example, digital models and document management systems emphasise the recording and management of the design’s tangible state, rather than the motivations, limitations and processes of team members. Given this preference, email is often the only means of promoting social presence within the team, but it is already heavily burdened and the level of interaction it enables is relatively limited.

3.1.3.ii. Inconsistent industry adoption and digital fragmentation

Internet-based communication is compounded by the inconsistent adoption of these tools and processes within the AEC industry (Ilich, Becerik & Aultman, 2006). It is common to find the full spectrum of digital skills and experience represented within a project team. This impedes digital collaboration because the different parties maybe using tools and processes that are poles apart in terms of technical complexity and capability.

Engsbo (2003) has identified a variety of reasons for the disparate adoption of technology within the industry (see Figure 3.3). Rectifying this problem is challenging because different issues exist at the individual, firm and network levels. In such an environment, technology adoption is a difficult and disjointed process, especially considering no industry body exists to dictate what its constituents should use. Additionally, Anumba (1998) points out that adoption has been hamstrung by industry researchers and software vendors, whose performance has been inconsistent in a number of areas:

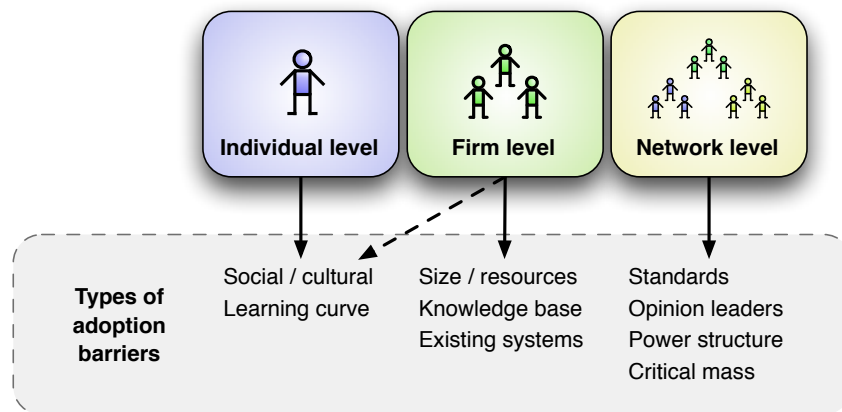
(Selected quotes from: Anumba, 1998, pp. 4-5)

- **Poor dissemination of research results** - Most academic research results are disseminated through conventional academic outlets – publications in journals and conference proceedings – and so do not adequately reach industry practitioners.
- **Poor marketing** - Not only has the marketing of these innovations lacked the aggression required in a conservative industry, but also it has often been based on the use of inappropriate and misleading terminology.
- **Inadequate user-interfaces** - Many early CAD and IT systems were victims of inadequate user-interfaces which blunted their appeal.
- **Mismatch between innovations and industry needs** - The low uptake of Construction IT innovations is the mismatch between some of the systems being developed by researchers and the needs of the construction industry.
- **Poor uptake by software developers** - Many software developers are conservative in their approach to new research prototypes based on new technologies.

The industry's project-centric nature magnifies these technology adoption issues. Excluding the contractual specification of CAD and document management formats, typically little high-level consideration is given to digital collaboration tools and processes when a team is conceived. This lack of strategic direction makes the deployment of complex and unfamiliar collaboration technologies challenging. In addition, the project-centric nature of the industry means that digital collaboration lessons are often forgotten when a team disbands, or found to be inapplicable under different circumstances.

Figure 3.3: Organisation levels and adoption barriers

(As found in: Engsbo, 2003, p. 459)



As a result of the industry's inconsistent and haphazard adoption of technology, project teams are now fragmented between the digital haves, and the have nots. The void between these two groups is showing no signs of closing, and if the current rate of technical evolution continues it will only grow larger. For example, within a typical project team the architect may spend the bulk of their time working on a sophisticated digital model, whilst the contractor's computer use maybe limited to the reading of email. This situation can dilute the potential of new technologies, because unless they are universally adopted their benefits will only be experienced by a select portion, and those who have not adopted them may be excluded from some design conversations.

3.2. The rise of the Building Information Model

The composition and role of the digital model within the project team has changed as the AEC industry has increased its use of digital collaboration processes. From humble beginnings as a replacement for the drafting table, the digital model has now evolved into the Building Information Model (BIM), a comprehensive, virtual representation of the proposed architecture. BIM is rapidly being adopted by the AEC industry because it offers a number of information management and process savings compared to its predecessor, Computer Aided Design (CAD). Unfortunately, the complex nature of BIM poses a number of collaboration impediments, that if not appropriately addressed can negate the benefits the technology provides.

3.2.1. The digital model as collaboration hub

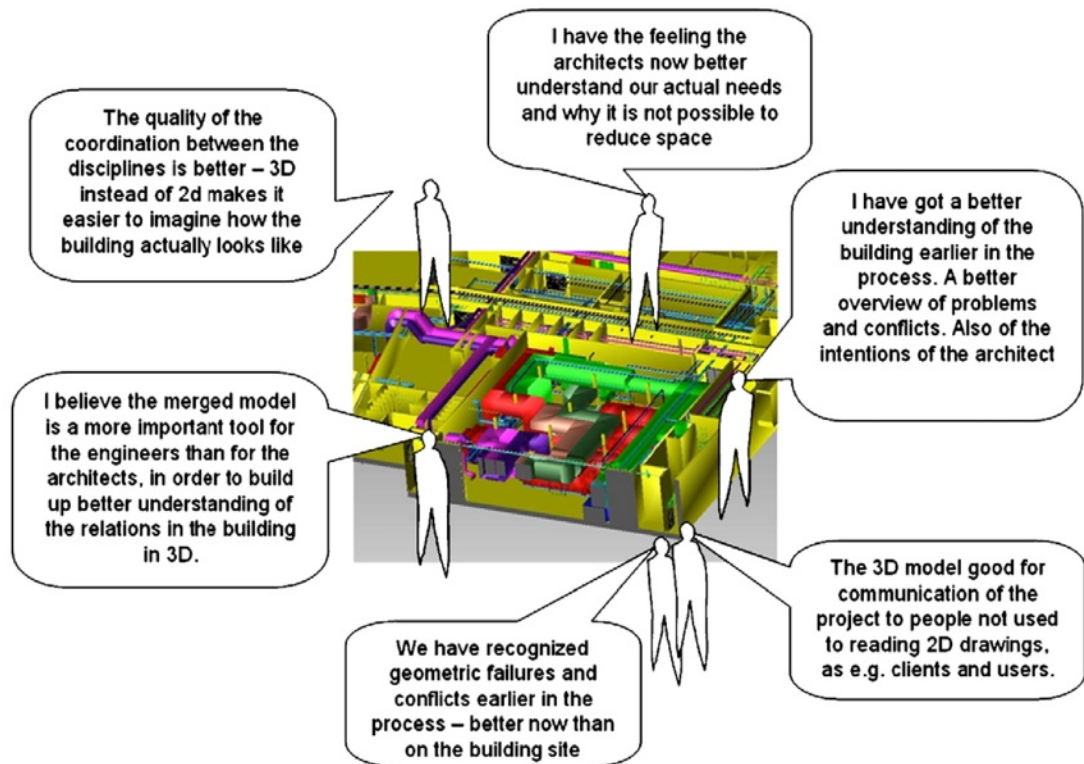
The centrepiece of a contemporary architectural project is the digital model because it stores the current state of the proposed architectural design. Within the AEC industry digital models are usually created with CAD or BIM software tools. Before project teams were linked by Internet-based communications tools, the digital model generally did not form an integral part of the collaboration process. Instead, it was treated as a standalone tool that was useful for generating printed documentation, and assisting in the visualisation and simulation of proposed designs. Its applicability as a collaboration tool was limited because of the difficulty associated in getting team members to interact with it in any meaningful way. For the majority, the closest they came to the digital model was reviewing the printed documentation that was generated from it. Those with appropriate software could be sent a copy of the model on physical storage media, but the associated time and cost overheads typically reduced the frequency and value of such exchanges. Even when these exchanges occurred, the ability for the parties to use these models as the basis for design collaboration was complicated by software interoperability problems (Alexander *et al.*, 1998, p. 2). Often this meant the recipient could not view the model at all, or they had little trust in what they saw because of display irregularities. Making matters worse, these interoperability issues usually ruled out making changes to the design represented within it.

Gross *et al.* observed that the rise of the Internet as the AEC industry's premier communication medium has led to a change in "the prevailing paradigm in computer-aided design, from traditional computer graphics involving a single designer interacting with a machine to construct CAD drawings and models, to a process of computer-mediated negotiation among multiple players" (1998, p. 466). High-speed Internet connectivity allows the project team to effortlessly interact with the digital model in a number of ways:

- The entire digital model can be exchanged via email or a document filing system.
- Static, generally 2-dimensional documentation, can be generated from the model and exchanged via email or a document filing system.
- Interactive experiences such as 3-dimensional walk throughs can be created and exchanged via email or made available on a project website/intranet.
- The model can be uploaded to a 'virtual world' environment where collaborators can interact with it individually or as part of a group.

By enabling more interaction with the digital model, “the function of CAD has expanded as a tool to communicate and collaborate as well as to better control all phases of the architectural practices” (Husin & Rafi, 2003, p. 509). Research by Moum (2010) has identified many benefits to collaboration when the digital model is used to foster understanding between team members. A selection of these described benefits are presented in Figure 3.4.

Figure 3.4: Collaboration benefits of the digital model (As found in: Moum, 2010)



In recent years the digital model has rapidly evolved from a simple drawing aid into a complex, comprehensive and semantically rich representation of the architectural design. This evolution was primarily driven by the benefits witnessed in other manufacturing industries who were using digital Product Models as a means of capturing and manipulating design information (Björk, 1995, p. 12). Providing further impetus for change has been the increased amount of information project teams were expected to manage as a result of their migration to Internet-based communication tools. Combined, these developments have brought about a significant change in the way digital models are perceived and utilised within project teams. This new generation of model is no longer the digital equivalent of an architectural drawing, but instead it is a semantically rich representation of the proposed architecture, and a highly structured and centralised repository for most project information. By making this change it is intended that

participants will be better able to understand the design's current state, and effect changes to it (Ibrahim & Krawczyk, 2003, p. 175). Given the digital model's role as a collaboration hub, this improvement should lead to more effective and capable project teams. In order to achieve these objectives, the AEC industry is now transitioning from traditional CAD tools, to a new generation of Building Information Model technologies.

3.2.2. What is a Building Information Model?

The Building Information Model (BIM) represents the latest thinking in how the majority of project information should be interacted with and recorded. At present the AEC industry is rapidly adopting BIM technologies in order to improve the capability, productivity and collaboration effectiveness of project teams. Unlike CAD, which at its core is a digital extension of the drafting table (Willis & Woodward, 2005, p. 72), BIM accurately records the analytical and semantic characteristics of an architectural design within a highly structured, semi-intelligent digital model. BIM is not a fundamentally new idea and draws much of its technical inspiration from Product Model technologies proven within the aerospace, shipbuilding and manufacturing industries (Eastman, Teicholz, Sacks & Liston, 2008, p. 28). This combination of CAD and Product Model results in an architectural information modelling tool capable of utilising semantic data structures to create efficient and versatile working environments (Ibrahim, 2006, p. 264). However, to attain these benefits the design team must consolidate all significant architectural information around a single, highly structured BIM.

As this is a relatively new concept, there is going to be debate as to what project information and technologies constitute a BIM. A broad, general view is that a BIM comprehensively describes the architecture's physical form and other related design and construction data (Amor & Faraj, 2001, p. 2). This ambiguity arises because the term was initially adopted for marketing purposes, rather than to clearly delineate a technology concept.

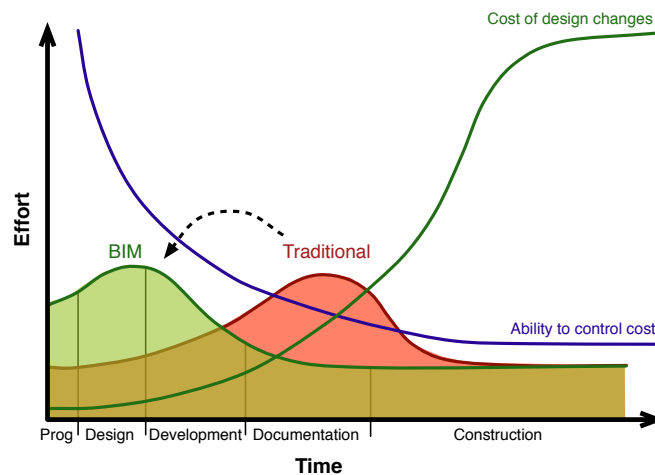
“After a bit of jawboning with Bentley's marketing folks, I got their agreement to use BIM as the top-level descriptive term for their latest design software. With Bentley and Autodesk both humming the BIM tune, we've covered well over 80% of the USA market (as defined by sales of current-generation 'CAD' tools). At an 80% tipping point, the dominos should fall into place for the rest of the market” (Laiserin, 2002).

Aided by this deliberately hazy definition, a general perception within the industry formed that BIM could be the panacea to its many problems, as Lamb, *et al.* observes:

“These days it's virtually impossible to make it through an architecture, engineering or construction event without hearing about BIM. Most people tout substantial cost benefits from adopting BIM. Some are even sounding the warning that those who fail to jump aboard the BIM bandwagon will be quickly left in the dust” (2009).

Beyond the hype, studies on BIM use within the industry have demonstrated significant process benefits and savings (Eastman *et al.*, 2008, p. 2). For example, industry case studies undertaken by Aranda-Mena, Crawford, *et al.* have shown that the majority of BIM users found their information management, process flow and sharing capabilities were improved (2008, p. 11). These case studies also highlighted that more effort is required in the initial stages of development of a BIM, compared to that of a traditional digital model (2008, p. 11). This occurs because BIM requires the early construction of a complex virtual model in order for the touted process benefits to be achieved. In contrast, traditional methods do not possess the functionality or semantics of a BIM, which leads to more effort being expended when design documentation is required. MacLeamy (2010) considers this redistribution of effort to be a positive influence on the project, as it encourages decisions to be made earlier in the process. He reasons that this leads to lower project costs and greater flexibility, because design changes are easier and cheaper to implement during this time. Figure 3.5 illustrates this relationship between time, effort and the associated cost of making design changes. This claim is supported by survey results from the University of Southern California, that found “41% of the respondents realised overall project profitability increase with the use of BIM, while 12% of the respondents reported that there was a decrease” (Becerik-Gerber & Rice, 2009). Likewise, “50% of the surveyed industry professionals found that overall project duration was reduced by up to 25%. An additional 13% of the respondents found that project duration was reduced between 25% and 50%” (2009). However, these results are clouded by the misconceptions that surround BIM, as many professionals consider it to be simply 3D CAD (Ibrahim, 2006, p. 263), whilst others have radically reorganised their project structures to attain such benefits.

Figure 3.5: The effort/time relationship between BIM and traditional digital modelling methods, relative to the cost of design changes (As found in: MacLeamy, 2010)



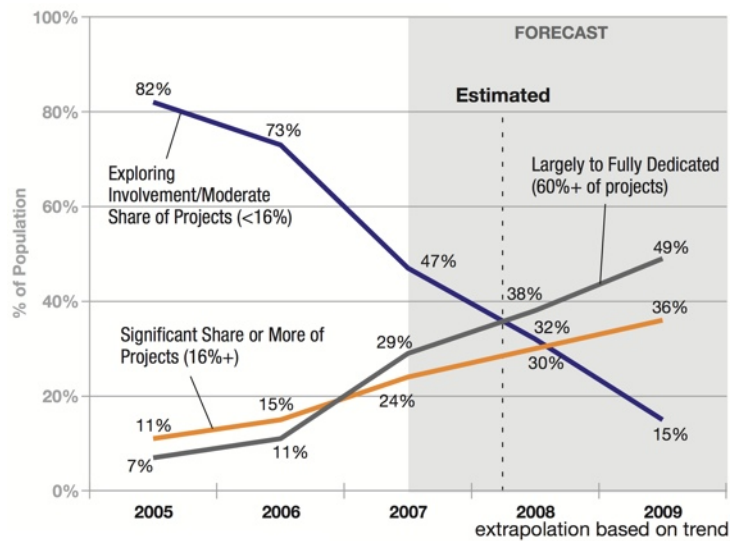
3.2.3. BIM in Practice

In a relatively short time BIM has established itself as an important industry concept and the natural successor to CAD. However, BIM is still a relatively immature technology that is yet to be fully integrated into the project team. To achieve optimum results with BIM, new collaboration and project delivery strategies are being pioneered. Unfortunately, the inability for most members to reliably access or exchange information stored within the BIM complicates these initiatives. Server implementations of BIM promise to address this problem, but they face a number of adoption challenges.

3.2.3.i. BIM on the desktop

BIM has gained a foothold within the AEC industry because it offers greater functionality and flexibility when compared to traditional CAD tools. These benefits are largely a result of the semantic data-structure it employs, which allows data stored within the BIM to be displayed and manipulated in a number of intelligent ways. Creating these semantic data-structures requires that users model architectural elements, rather than simple lines and shapes. Given the complex nature of this modelling activity, current desktop BIM implementations are difficult to use, primarily targeted at those responsible for shaping the design, such as architects, technicians and engineers. From the perspective of the architects especially, this exclusivity is considered desirable because it affords them “the opportunity to ‘deal themselves back in’ to the knowledge management of a project from beginning to end and beyond” (Ambrose, 2006, p. 186). Unfortunately, for the rest of the team who lack the software and skill-set necessary to access the BIM, they must rely on being supplied with data exported in a digital format they can utilise.

Figure 3.6: The adoption of BIM within projects
 (As found in: Young, Jones & Bernstein, 2007, p. 11)



A recent Erabuild survey found that “BIM is used in around 20% of projects for architects and in around 10% of projects for engineers and contractors” (Kiviniemi, Tarandi, Karlshøj, Bell & Karud, 2008, p. 79). Although a relatively small percentage, trends indicate that BIM adoption will continue to steadily increase (see Figure 3.6). Studies have found that when BIM is successfully applied within a project it generally results in significant process efficiencies (Lamb *et al.*, 2009). Unfortunately, migrating to BIM is currently a demanding financial and resource investment because the concept, tools and associated processes are very different to that of CAD. These costs pose a problem for organisations with limited resources, because the industry has been unable “to pass on the costs of the implementation and use of BIM to clients, either through fees or direct owner participation” (Becerik-Gerber & Rice, 2009, p. 5). However, assuming BIM follows established technology trends, these implementation costs should decrease as competition within the software market grows and knowledge of the technology matures.

From the perspective of collaboration, a desktop BIM provides few immediate advantages over conventional, CAD-based methods because the majority of the team does not interact with it directly. Nevertheless, in the hands of a skilled user, the modelling and process benefits associated with BIM can improve the flow of information between team members. This occurs because the information within a BIM is better managed, design changes are easily worked into the model, and documentation can be efficiently generated. However, attaining these benefits requires the BIM to be developed to a high-standard at the earliest possible stage of the design process. In many projects this can be difficult because the

resources and relationships necessary for this development to occur are not available until later in the process. Additionally, the inability of most of the project team to interact with a desktop BIM can pose a collaboration bottleneck, because they must rely on a few skilled intermediaries to extract information from the BIM and record contributed design changes. To address these process shortcomings, project delivery strategies are being explored that can maximise the value of BIM, and optimise its collaboration benefits.

3.2.3.ii. *BIM within the team*

BIM is a powerful technology concept, but to achieve optimum results it needs to be applied within a team that is working closely together from the outset. Fortunately, “many firms are already moving toward more collaborative teams, especially with the expanded use of design-assist and design-build on projects” (Young *et al.*, 2007, p. 4). Integrated Project Delivery (IPD) is the most recent and prominent example of such a delivery strategy. By leveraging new technologies such as BIM and Internet-based collaboration, IPD strives to get all parties “involved on a project as early as possible—ideally during schematic design—to provide a collective expertise to the development of a project before anything is designed” (Integrated Project Delivery, 2009). BIM plays a central role in IPD because it provides the cohesive, centralised digital model that is necessary to orchestrate the actions of a closely knit team. In return, IPD benefits BIM because it enables the early formulation of the project team and the rapid development of the architectural design. These factors ensure more time and resources are invested on the BIM during the project’s initial stages, which is essential for it to deliver its promised process benefits.

IPD has demonstrated that it can achieve successful results at the project and collaboration level (Matthews & Howell, 2005, p. 51), but it is yet to be broadly adopted within the AEC industry (Kent & Becerik-Gerber, 2010). For many IPD is a difficult proposition because it requires extremely close working relationships, along with the pooling of the project’s risks and rewards (IIA, 2007, p. 5). Whilst these are important characteristics of a healthy collaboration environment, for many within the segmented and conservative AEC industry these are foreign ideas that will take time to be accepted. For teams that do implement BIM-friendly delivery strategies, they must still overcome a number of technical challenges if they are to succeed. The most challenging of these is that with current, desktop BIM tools, managing changes from multiple people is a complex and often unreliable process.

For any collaboration system centred around a BIM to operate effectively, participants must be able to reliably interact with the data it possesses. The most straightforward means of achieving this is through the exchange of digital files that read from, or contribute to, the central BIM. This can be difficult in practice due to the incompatibilities that exist between different software tools and data formats. Digital interoperability is a long-standing problem within the AEC industry (Froese, Grobler & Yu, 1998). A large amount of research has occurred in this field, which has led to the development of the Industry Foundation Classes (IFC) (Kiviniemi, 1999). As described by Froese, *et al.* “IFCs are used to assemble a project model in a neutral computer language that describes building project objects and represents information requirements common throughout all industry processes” (1999, p. 18). Although the majority of BIM tools support the IFC standard in some form, research has found that “the certification process allows poor quality implementations to be certified and essentially renders the certified software useless for any practical usage with IFC” (Kiviniemi *et al.*, 2008, p. 79). Given the inconsistency of the ‘standard’ data format, exchanging data between different models can be a very unreliable and inefficient process unless both parties are using exactly the same software (Young *et al.*, 2007, p. 4). To avoid manually exchanging files, many BIM tools have functionality for allowing multiple people to interact with the same digital file. However, this functionality is often found to be missing key features, too complicated, or unreliable in practice. To address these shortcomings, researchers and software vendors are transforming BIM from a single-user desktop tool, into a server-based system that can be reliably and simultaneously accessed by the entire project team.

3.2.3.iii. BIM on the server

To exploit the full potential of BIM all members of the project team must be able to interact with a centralised and cohesive digital model. This is often difficult with desktop BIM tools because they are primarily designed for one person to use at a time. To address this shortcoming, researchers and software vendors have developed dedicated BIM servers (Adachi, 2001), which employ a client-server architecture to provide the entire team with regulated access to project data. Instead of directly interacting with a single BIM file, the server daemon acts as an intermediary who manages access, exposes data in a variety of formats, and validates the changes received from software clients. This approach supports more effective collaboration because team members can be confident of the currency and

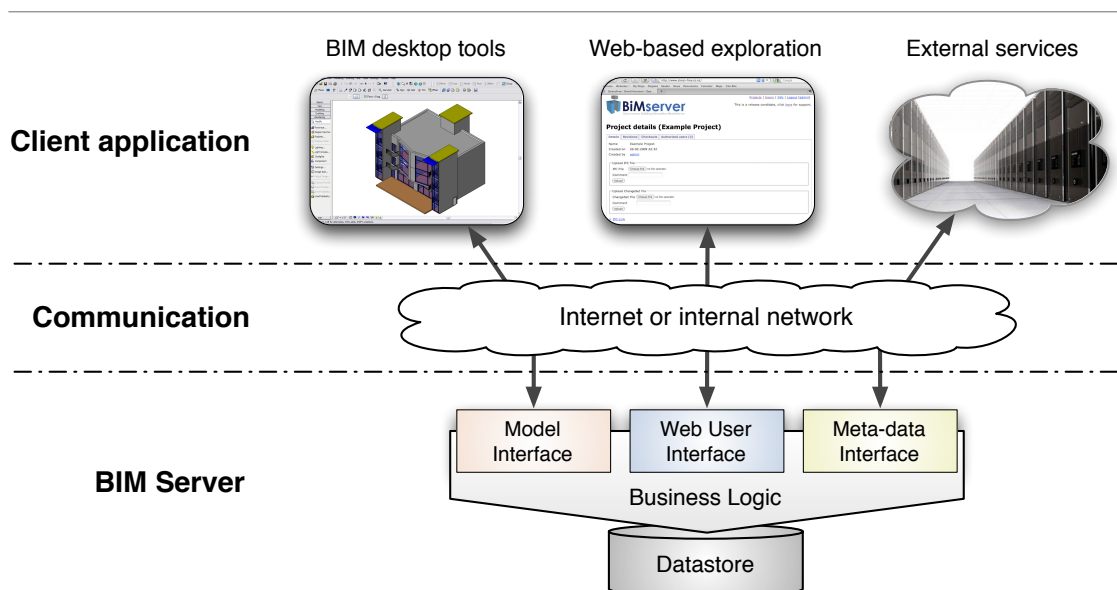
validity of the project BIM, and interoperability problems are eased as only small packets of data are exchanged between the clients and server.

The greatest benefit of a BIM server is that the model is able to exist as an independent and dynamic entity, separate to that of the applications which interact with it. This freedom greatly expands the number of ways by which project data can be accessed and modified (see Figure 3.7):

- Team members with access to desktop BIM tools can undertake complex alterations to the model, generate documentation, or perform simulations.
- Team members without desktop BIM software can use a web browser to determine if changes have been made to the model, extract relevant documentation, or contribute simple information such as notes and images.
- External services can automatically extract or modify data within the BIM in order to notify third-parties of changes, index the model for search purposes, or to update meta-data such as material pricing and availability (Harrison, 2009).

In combination these interactions result in a more effective collaboration environment because they transform a static, idealised BIM into a dynamic, virtual member of the project team. These different interfaces also ensure that all authorised members of the team can access the BIM using tools and services that are appropriate for their needs.

Figure 3.7: The interfaces of a BIM server and the clients which interact with them



BIM servers are a powerful concept with a great deal of potential, but they have yet to be widely adopted within the AEC industry. For most organisations it is a difficult proposition because it overturns decades of investment in file-based workflows, and many have only recently made a commitment to desktop BIM. Further hindering adoption is the immaturity of the BIM server market, which is yet to see strong and sustained investment from the major AEC software vendors (Harrison, 2009). Given these challenges, it is unlikely that usage BIM servers will grow in the foreseeable future. This is unfortunate as it could potentially resolve many of the collaboration shortcomings inherit within desktop BIM.

3.3. Digital collaboration barriers in a BIM-centric team

The Building Information Model is gaining widespread industry adoption, and will soon be a permanent fixture of most project teams. This has occurred because it has proven itself capable of delivering process efficiencies and consolidating architectural information. However, current implementations of BIM pose a number of collaboration impediments that can impair the team's ability to record and understand the project's issues, events and decisions. The complications that arise when multiple parties interact with a single BIM commonly lead to multiple digital models being concurrently developed during a project. This has negated a core premise and benefit of the technology, namely that it can act as a cohesive and comprehensive description of the architectural response. Combined, these issues limit access to timely and relevant information, because the project's design decisions and events cannot be easily recorded, communicated, or understood within this fragmented digital environment.

3.3.1. The collaboration shortcomings of BIM

BIM provides segments of the project team with significant functional and process benefits. However, its complexity and rigid nature establishes high barriers to entry which can hinder the flow of relevant information throughout the project team. The accurate and pristine image presented by BIM can also pose collaboration problems when it comes to reflecting onsite activities. Architecture is almost never built exactly to plan, but recording these variances is challenging because the parties responsible for construction are generally unable to contribute directly to the BIM.

3.3.1.i. Complexity reduces participation

If team members cannot access or contribute to this process then the outcome of the project can be put in jeopardy (Lamb *et al.*, 2009, p. 7). Participation is an important factor in collaboration because architecture is the physical representation of a collective decision making process (Cooper *et al.*, 2005, p. 125). BIM imposes process and knowledge barriers to participation due to its reliance on a single, complex data structure. In an effort to ensure the digital building model's integrity, the authority to manipulate the data within it is restricted. Even when permission is granted, few team members understand and are capable of using the complicated software interfaces which govern the building model.

The participation bottleneck formed by this complexity means the project team generally relies on selected participants to funnel relevant design data and decisions into the BIM. This role of digital shepherd generally falls to the architect, owing to their status in the team, and close association with the digital building model. Many architect's appreciate this move because it reinstates them into the role of master craftsman. (Barrow, 2004, p. 102) Unfortunately, those who take on this role can consciously or subconsciously filter out information that is relevant, or even vital, to the comprehension of design issues and decisions. Given the preeminence of BIM within the team, this can lead to decisions being made based on false or outdated information, whilst the unrecorded contributions maybe lost entirely.

3.3.1.ii. Rigid centralisation leads to the filtration of information

It has been acknowledged by software vendors that “the building life cycle is a demanding beast and requires a careful mix of tightly coupled and loosely coupled data structures” (Bentley, 2003, p. 8). BIM typically uses a rigid data-structure which limits the type and quantity of data that can be stored within the model. A rigid structure allows BIM to have semantic meaning, which means it can be consistently displayed and intelligently manipulated. However, problems arise when information falls outside of this structure because it breaks traditional conventions or is beyond the BIM's functional capabilities. To fit it within the confines of the BIM, such information often needs to be editorialised and associated with a foreign semantic system. This categorisation process can degrade the recorded information, reducing its usefulness during collaboration exchanges. AEC researchers and software vendors are aware of these structural limitations, and are continually extending the capability of their products and data formats (Amor, Jiang &

Chen, 2007, p. 160). Unfortunately, these extensions typically increase the complexity of the BIM, and take place with the knowledge that no rigid structure is capable of handling all of the project team's potential data and semantic needs.

The centralisation of project data within a BIM simplifies information management, but determining where the line is drawn between what is 'in' or 'out' of the model is difficult. A range of professions and interests are represented within the project team, all of whom have specific information needs and design concerns. Given this situation, BIM vendors are faced with a challenging dilemma, because they are tasked with storing the project's data, but they cannot store so much that the experience becomes unwieldy. Additionally, any data stored must be awarded some level of significance so that it can be emphasised to specific interest groups. For example, architects are primarily interested the design's form and its spaces, whilst engineers its services and structural system. These software design decisions can impair project collaboration because important information may inadvertently fall outside of the BIM, or be awarded too little significance relative to its role within the design. Accommodating these different needs has led to the bifurcation of the BIM market along professional lines (Kiviniemi, Fischer & Bazjanac, 2005). This has resulted in projects using multiple, independent BIM and CAD files which are tailored to meet specific interests. Figure 3.8 illustrates the variety of models that can emerge when this strategy is applied, and the complex processes which are required to shepherd and synchronise project information between them. Whilst this proliferation of digital models satisfies the needs of specific groups, it undermines many of BIM's underlying concepts and benefits, and makes accessing relevant information a more complicated process.

Figure 3.8: The use of multiple digital models within the project team and the complex information flow which occurs as a result (As found in: Moum, 2010, p. 4)

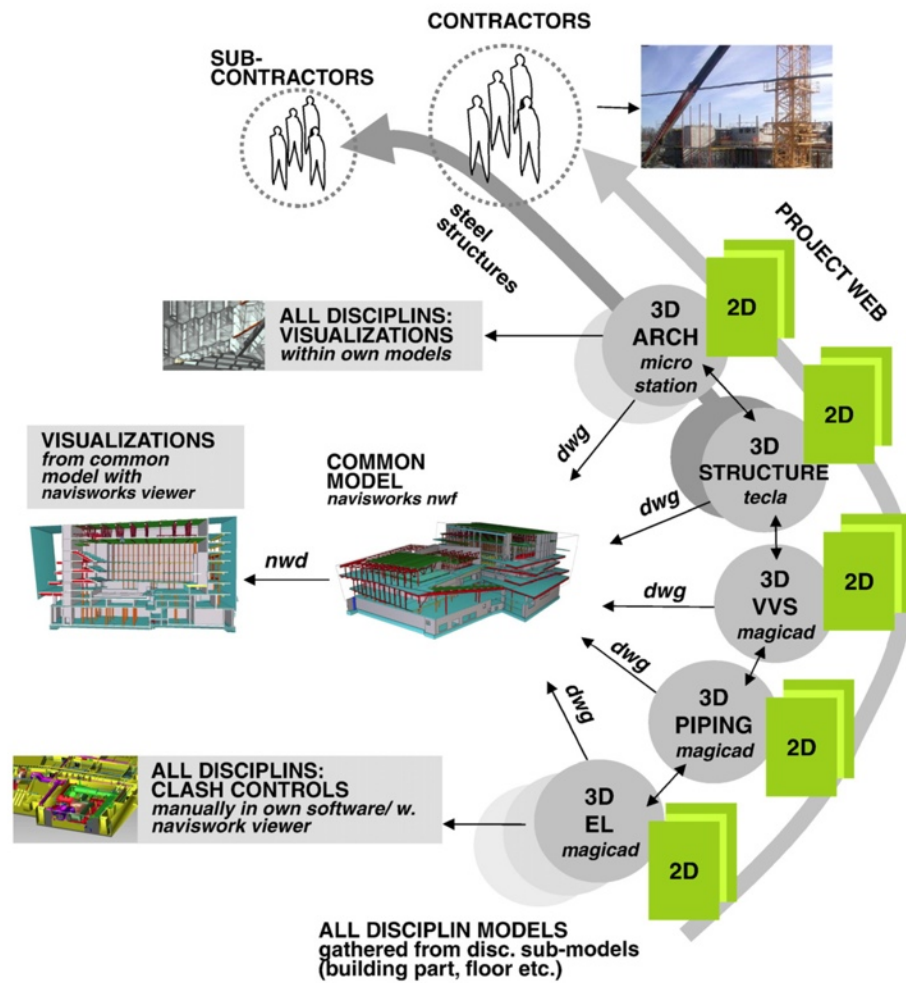


Fig. 4. The 3D object model system, software and applications summer 2007 (detailed design phase).

3.3.1.iii. Virtual accuracy confuses practical reality

Accuracy within an architectural project is crucial, but it is equally important to know where inaccuracies and tolerances lie. Architecture ultimately manifests itself as a physical entity, and it is important for the project team to understand where, how and why the physical form deviates from its virtual blueprint. Traditional design representation depicted an abstract and partial description of the intended built form. In contrast BIM's capacity to depict a highly accurate, yet ultimately idealised, virtual truth risks impeding the ability of design participants to comprehend or accept the discrepancies between the virtual and physical realms. This is an issue that becomes pronounced as rapid design changes and construction inconsistencies are introduced into the process. If those administering the BIM cannot keep pace with these changes then information will be lost, incorrect decisions made and the overall understanding of the design collaboration process will suffer.

3.3.2. Digital fragmentation and the need for timely and relevant information

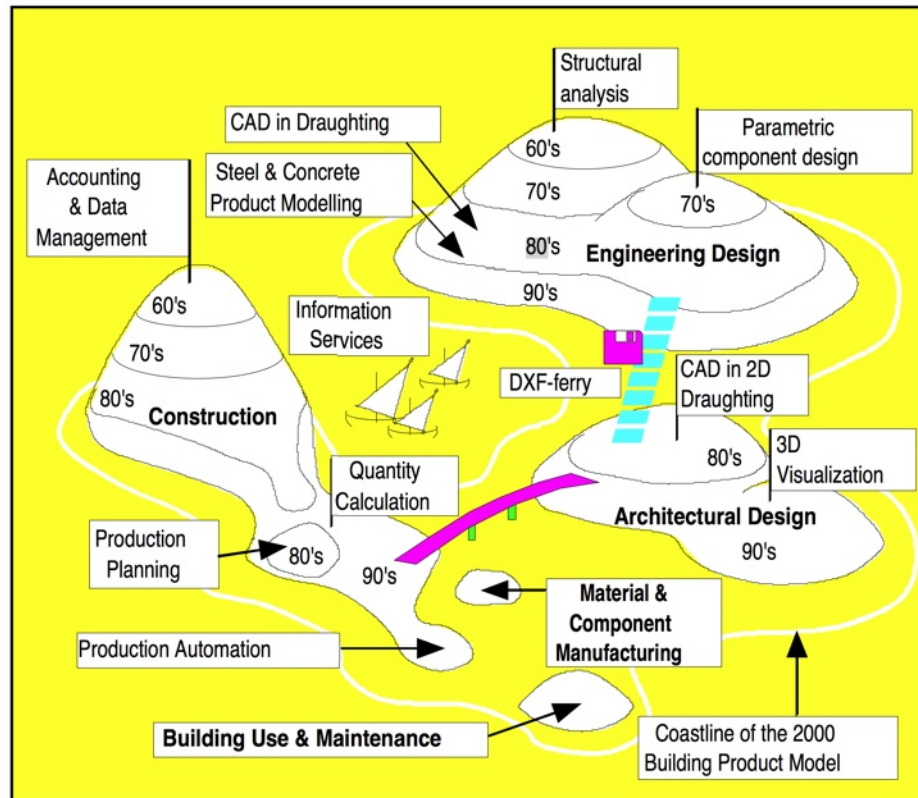
The growing level of digital fragmentation within the project team is a significant barrier to effective collaboration. There are many practical and technical reasons for this digital fragmentation, such as:

- **Functionality** - needs vary depending on the services team members provide.
- **Knowledge** - the level of technical skill and experience team members possess.
- **Resources** - the ability of participants to acquire new technology and alter processes.
- **Access** - the ability of participants to utilise technologies at their workplace.
- **Variety** - the quantity, quality and diversity of technologies within the market.
- **Innovation** - the pace of technology development can quickly obsolete tools.
- **Competition** - technology use can be restricted to provide a business advantage.

Combined, these issues have resulted in a diverse technology landscape within the project team. Within this environment, the majority of a project team's information is concentrated around isolated digital pockets that are unevenly distributed between team members. Figure 3.9, the "islands of automation" (Björk, 1995, p. 11) is an excellent graphical representation of this landscape and the digital stratification that exists.

The functional disparities and technical incompatibilities within project teams have led to digital fragmentation. The effectiveness of architectural collaboration is impaired by this fragmentation because it is difficult for participants to access timely information, or participate in relevant design conversations. The graphic in Figure 3.9 was prepared during the 1990s, and it reflects the opinion held by many at that time that the Building Product Model, now more commonly known as BIM and the IFCs, would form a permanent land bridge between these islands of digital information. Unfortunately, BIM has so far proven incapable of filling this role, and in many situations it has only added to the team's digital fragmentation. Whilst it is possible that an implementation of BIM will emerge that can unify the project team and heal these digital divisions, in the foreseeable future this is unlikely. For example, server-based BIM is a promising development that addresses many of BIM's current collaboration shortcomings, but the technology is immature, and numerous adoption barriers must be overcome if it is to succeed.

Figure 3.9: The digital fragmentation caused by the technologies used within the project team (As found in: Björk, 1995, p. 11)



Healing this digital fragmentation will require a new, more permeable layer of information, where all participants can contribute to the ongoing design conversation and the processes associated with it. In this regard, the widespread adoption of Internet-based communication tools has been a promising development, but at present these have not been used to explicitly address this problem of digital fragmentation. Instead, they have primarily been used to reduce collaboration friction by enabling faster, higher-bandwidth, more frequent and cost-effective digital interactions. Unfortunately, removing this friction has multiplied the team's information management problems and generally led to further digital fragmentation. To address this problem, the focus needs to be shifted to leveraging the concepts inherent within the Internet to create digital tools that allow the team to collaboratively record and convey the project's Building Story. These common threads of information would help to bridge the isolated pockets of data that currently exist within the team. The intention of creating a digital Building Story is not to expose all of the team's data, but to allow participants to better understand the project's current state, and its design issues, requirements and decisions. Providing the team with this knowledge will lead to more effective digital architectural collaboration, because participants will be better able to access, identify and contribute timely and relevant information.

3.4. Summary

The widespread adoption of Internet-based communication tools has fundamentally changed the way project teams collaborate within the AEC industry. This digital collaboration migration has significantly increased the speed and quantity of data that is communicated during a project. Whilst this has enabled more flexible and distributed project teams, it has often impaired access to timely and relevant information because participants are unable to process this sudden influx of digital information. These shortcomings have been partially addressed by the Building Information Model (BIM), which is a comprehensive virtual representation of the proposed architecture. When appropriately applied within a project, BIM's concise and semantically rich data-structure often allows for a more flexible and efficient design process. Unfortunately, the complex nature of BIM, coupled with the unequal distribution of digital knowledge and resources with a project, means that very few members of the project team can directly contribute to the digital model. This lack of accessibility increases the digital fragmentation within the team and makes it difficult for participants to accurately record their design decisions.

The project team's digital landscape is fragmented because of the rapid, uneven and relatively unplanned adoption of technology by the AEC industry. This fragmentation complicates the digital collaboration process, and can often lead to team members being excluded from design conversations. To address this fragmentation, the team's digital collaboration environment needs to support the recording, organising and retelling of the project's Building Story. Through the telling of these stories, participants will gain a better understanding of the project's current state, and its design issues, requirements and decisions. This will not heal the digital fractures within teams, but it will enable the construction of narratives that bridge the project's isolated pockets of information. To achieve this goal, the next chapter describes Hyperlinked Practice, which is a hypothetical, Internet-centric collaboration environment that facilitates the telling of Building Stories. Through the adoption of Hyperlinked Practice, collaboration will be more effective because participants will be able to access timely and relevant information.

4. Improving Collaboration Through Hyperlinked Practice

Constructing digital Building Stories to bridge the gap between BIM and the team

Building Stories promote understanding within the team because they allow participants to reflect upon a project's current state, and the requirements, issues and decisions which contributed to its design outcome. This understanding enables participants to access and contribute more timely and relevant information, which in turn leads to more effective collaboration. Efficiently constructing Building Stories is currently difficult because the digital fragmentation of project teams, coupled with the trend of consolidating project information within Building Information Models (BIM), has resulted in the design process being digitally recorded in an inconsistent, inaccessible, and often obscure manner. To address this problem, the architecture, engineering and construction (AEC) industry needs to adopt the digital collaboration concept of Hyperlinked Practice, which is a Web-centric means of perceiving of, communicating and storing a record of the design process.

Hyperlinked Practice promotes the inclusive and comprehensive recording of the design process by digital tools and processes that enable Building Stories to be efficiently constructed and reflected upon. The emphasis of Hyperlinked Practice is on capturing and forming relationships between the design requirements, issues, problems and discussions that contributed to an architectural outcome. To achieve this, Hyperlinked Practice proposes that the information within a project be perceived as a distributed and interconnected cloud of information that any member of the project team can contribute to. In this sense, Hyperlinked Practice aims to embody many of the characteristics which have made the World Wide Web a successful, all purpose medium for communication and collaboration. Given this objective, Hyperlinked Practice shares many concepts and technologies in common with the Web. Like the Web, the technologies that facilitate Hyperlinked Practice will evolve and change over time as new collaboration challenges and opportunities emerge. Given the rapid pace of information technology innovation, it is therefore important that a set of fundamental principles for Hyperlinked Practice be established, because this will help to guide the development of digital tools that facilitate this environment today, and in the future.

4.1. Putting Building Stories into architectural practice

Building Stories record, organise and present a project's events, activities and artefacts in a manner that allows team members to improve their understanding of the overall architectural process and the personalities who shaped it (Martin *et al.*, 2003, p. 30). At present there is no standard method for recording, organising and experiencing a project's Building Story. Partly this is due to the concept's immaturity, but to a greater extent it reflects the varied and at times haphazard manner people choose to review and record the problem solving process they have undertaken. This problem is not unique to Building Stories, for example in the related field of Case Based Reasoning the "development of case memory [the representation of the previously solved problems] is an ill-defined process that can be as difficult as the knowledge acquisition stage in developing an expert system" (Maher, Balachandran & Zhang, 1995, p. 6). Therefore, any processes that are put in place to record the Building Story must be as flexible as possible, in order to accommodate the multitude of ways the underlying architectural design process can be undertaken, recorded and reviewed.

When conducting research on Building Stories, Martin, Heylighen and Cavallin applied a retrospective, two stage process to construct Building Stories for a series of architectural projects in San Francisco (2005). The development of these Building Stories involved industry professionals from the respective project teams, and student researchers who conducted the bulk of the recording and analysis. The first stage of this process was a series of group meetings conducted over a seven week period. These were held to establish an overview of the project from the perspective of the professionals involved. During this time the students grouped the project's events and information into six lines of investigation (Martin *et al.*, 2003, p. 33):

- **Requirements** - The project's definition and clients' aspirations.
- **Process** - The marketing process, project team organisation and work plan.
- **Design** - The design process from schematic design to construction documents.
- **Construction** - Project construction management and administration.
- **Completion** - Commissioning, measuring success and post occupancy evaluation.
- **Innovation** - Examples of practice innovations.

Although time consuming, this collective process ensured everyone involved gained a broad understanding of the project and its events. Through analysing this historical overview, significant project events were identified that warranted further investigation (Heylighen *et al.*, 2005, p. 421). The second stage of Building Story development shifted to “putting flesh on the skeleton, i.e. formally constructing the story details” (Heylighen *et al.*, 2005, p. 421). Here, industry professionals from the respective project teams elaborated on key events, while the students collated and organised this information into a series of written reports that described:

- the project’s significant events;
- the activities that took place during them;
- the artefacts that were generated as a result;
- and the influence all of these things had on the project and those involved.

Combined with the overall history established in stage one, these written reports helped to form a cohesive and relatively comprehensive Building Story. To share this assembled knowledge, the finalised Building Stories were published to the Web, where they could be easily accessed by interested parties, or indexed by Internet search engines.

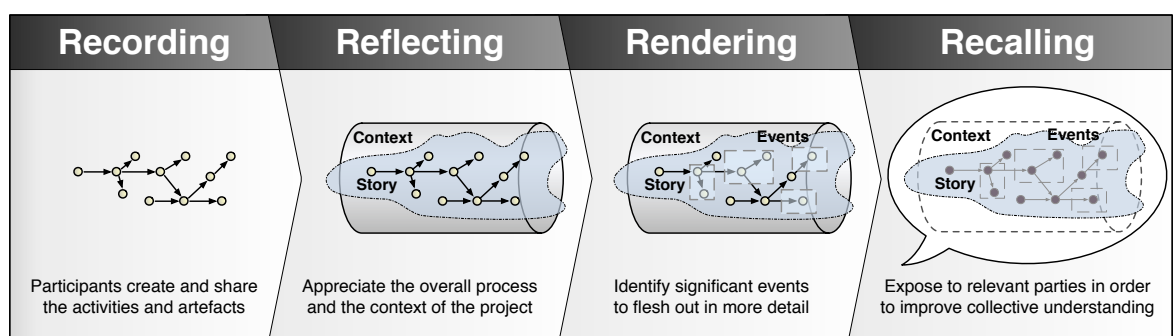
4.1.1. Recording, reflecting, rendering and recalling

The Building Story development process followed by Martin, Heylighen and Cavallin was for research purposes and not intended for widespread industry application. However, the method’s four underlying development phases (illustrated in Figure 4.1) can be used as the basis for methods that are more applicable to a typical project team:

- **Recording** - The team records and collates the actions and artefacts related to the project. In the case of the academic research, students met with project team members to record the overall process they had followed from the project’s conception through to its completion.
- **Reflecting** - The team reviews the assembled collection of actions and artefacts in order to broadly categorise the information and gain a general understanding of the overall project. In the academic example, students prepared a written report and presented their findings after seven weeks of study.

- **Rendering** - Team members select important or interesting aspects of the project and elaborate on what occurred, what was learnt and who was involved. In the academic example, the students developed this understanding during the second half of the semester in conjunction with members of the project team.
- **Recalling** - The overall Building Story, which includes its component events, activities and artefacts, is presented to relevant parties in a manner that can be easily consumed. In the academic example, the students published their compiled Building Stories to a public website where anyone with an internet connection could access it.

Figure 4.1: The four phases of Building Story construction and utilisation



For the purposes of simplicity and completeness Heylighen, Martin and Cavallin undertook these four phases sequentially and after the projects had been completed. However, if the Building Story is to be of benefit to collaboration, participants must develop and utilise it whilst the project is progressing. To achieve this the Building Story development process must be streamlined and an integral part of the team's workflow.

4.1.2. Constructing Building Stories within project teams

Like the architectural design process it portrays, the exact method a team uses to record, reflect, render and recall a Building Story will vary depending on the project's conditions. Irrespective of the exact method applied, a common set of requirements must be satisfied so that Building Stories can be recorded and utilised by the team. Similarly, common challenges must be addressed during each phase of the Building Story's development. These functional requirements and challenges are outlined in Table 4.1.

Table 4.1: Requirements and challenges of Building Story development

Recording	
Functional requirements	Practical challenges
<ul style="list-style-type: none"> • Capture - Team members should document the activities and the artefacts that are related to the project. • Publish - For the Building Story to grow, participants must be able to distribute recorded details to relevant team members. • Collate - Participants need to bring information from the various sources together so that it is cohesive and manageable. • Store - The resulting information needs to be reliably stored over a long period of time. Maintaining multiple copies of the information will ensure these contributions will not be lost when team members depart. 	<ul style="list-style-type: none"> • Pressure - The majority of project teams are short on time and resources. The time committed to developing Building Stories will be restricted and the outcomes terser. • Quantity - The project's activities and artefacts need to be recorded and distributed, but this cannot occur in a manner that overwhelms or confuses participants. • Legacy - The team's existing organisational and reporting structures must be respected. Breaching these relationships may cause confusion, expose confidential information, or be legally comprising. • Reluctance - Due to the litigious nature of the industry, many will be unwilling or unable to record sensitive information.
Reflecting	
Functional requirements	Practical challenges
<ul style="list-style-type: none"> • Review - Participants must understand what information has been contributed to the Building Story at both macro and micro levels. This will allow significant events, relationships and trends to be identified. • Projection - Many external forces will shape the evolution of the project. It is important that these contextual events are presented alongside contributed information when the Building Story is being reflected upon. • Categorise - As participants reflect on the recorded information, they need an efficient means of grouping related topics. Depending on the viewpoint of the participant, the same information maybe categorised differently. 	<ul style="list-style-type: none"> • Accessibility - The information that is recorded will exist in a variety of locations. Some participants may have limited access to parts of this repository due to organisational or technical barriers. • Diversity - A project's recorded information will span a broad range of topics and media. Many participants may lack the knowledge or technical ability to consume much of the information presented. • Misunderstandings - Information related to activities and artefacts can be misunderstood and miscommunicated. This can stem from the complexity of the subject matter, technical issues and resource pressures.

Rendering	
Functional requirements	Practical challenges
<ul style="list-style-type: none"> • Relate - Events form when meaning is applied to a series of related activities and artefacts. To achieve this, participants need to be able to construct tangible relationships between these entities. • Summarise - Participants need to be able to describe why a series of activities and artefacts constituted a significant event, and what the ramifications of this were. • Discussion - Opinions on how an event unfolded and why it is significant may vary. Recording this discussion is valuable because it serves to reflect the different viewpoints and goals of team members. 	<ul style="list-style-type: none"> • Participation - Given the broad range of knowledge and skills team members possess, ensuring all relevant parties can contribute to the rendering of events could prove difficult. • Bias - If a single party yields too much influence on the Building Story, then the events recorded may present a slanted or fabricated view of the project's evolution. This rewriting of history could have serious practical or legal ramifications. • Repetition - Teams lack the time and resources to comprehensively describe a project's events. To compensate, information that has already been recorded needs to be reused as much as possible.

Recalling	
Functional requirements	Practical challenges
<ul style="list-style-type: none"> • Preserve - The Building Story should remain accessible, cohesive and legible well after the building has been occupied. • Browse - Interested parties need to navigate through the Building Story. The information must be structured to allow it to be viewed from a variety of perspectives. • Search - People expect to quickly find relevant information. To achieve this, an index is required which spans the breadth of the Building Story's knowledge base. • Share - Building Stories are of little value if the information within them cannot be exposed to interested parties. The majority of these interested parties will not be members of the original project team. 	<ul style="list-style-type: none"> • Decay - As the team changes there is a strong risk that project information will be lost. This risk intensifies once the building is complete and the design team disbands. • Relevance - Information about the design and its built-form can become irrelevant or misleading over time. This needs to be acknowledged, but often the contributor of the information has left the project. • Confidentiality - The mechanisms used to expose a Building Story should enforce restrictions for privacy and legal purposes, otherwise participants may not be comfortable contributing knowledge.

4.1.3. Overcoming the barriers to Building Stories

Emphasising the recording, organising and experiencing of the Building Story poses a series of legal, process, organisational and technical challenges. The AEC industry is a fragmented and litigious environment, thus attaining the interaction and integration levels required to construct a successful Building Story is not an easy prospect. To overcome these challenges, it is best to focus on the tangible problem of how the digital collaboration tools used within the industry can help project teams record, organise and experience Building Stories. Establishing the digital environment necessary to efficiently record Building Stories will allow their collaboration benefits to be demonstrated, which in turn will provide measurable incentives for overcoming the legal, resource and organisational barriers that may exist within project teams. These measurable incentives include:

- Explicit examples of the design iterations within a project, how these were recorded, who reflected upon them, and what benefit this brought to the overall design process.
- A clearer understanding of the time and resource investment required to digitally compile Building Stories during real world architecture projects.
- Case studies that illustrate the time and cost savings possible when a comprehensive record of the design process is maintained throughout a project.

4.2. Constructing Building Stories through Hyperlinked Practice

Constructing Building Stories within industry project teams requires a comprehensive and inclusive record of the design process that can be reflected upon by the entire team. Unfortunately, the current digital collaboration environment is not conducive to creating a comprehensive record of the design process, due to the digital fragmentation within teams (see Section 3.3.2), and the consolidation of project information within complex Building Information Models (see Section 3.3.1). To address this problem, the industry needs to embrace the concepts of the World Wide Web to facilitate Hyperlinked Practice, which is a distributed and interconnected digital collaboration environment that is conducive to the recalling, reflecting, rendering and recalling of a project's Building Story.

4.2.1. Embracing the Web to bridge the digital fragmentation of the project team

Efficiently constructing and reflecting upon Building Stories requires a comprehensive digital record of the architectural design process. Unfortunately, the digital fragmentation

of the project team, coupled with the trend towards consolidating project information within Building Information Models, has led to the design process being recorded in an inconsistent, inaccessible and often obscure manner. The concepts and technologies of the World Wide Web stand as a potential means of enabling the entire project team to digitally record and reflect a project's Building Story. The Web is the largest and most flexible distributed information network in existence. Berners-Lee and Cailliau (1990) put forward the proposal for what would become the World Wide Web in 1990, and fifteen years later the Web had grown to encompass over 11.5 billion web pages distributed across more than 70 million websites (Gulli & Signorini, 2005). The Web's success stems from the ability of anyone with an Internet connection and web browser to read, interact with, and publish new content to it. The rapid and broad growth of the Web has seen it influence all aspects of modern life and business. The AEC industry has not ignored the Web, but "despite the obvious benefits and explosive growth of Internet usage in many areas of business, the AEC industry has not completely realised the benefits of web-based project management and collaboration technology" (Becerik, 2004, p. 239). Architectural collaboration stands to be made more effective if the inclusive, flexible and decentralised characteristics of the Web were to be embraced by the AEC industry, and embodied within the digital collaboration tools used by its project teams.

The availability of high-speed Internet connectivity has revolutionised communication within project teams (see Section 3.1). Email is a prominent enabler of these digital interactions (Vollmer & Gaßner, 2005, p. 124), but many organisations and project teams have deployed web-based intranet, extranet and document management systems to promote and facilitate collaboration between internal staff and external members of the team (Wilkinson, 2005, p. 27). But as Ingirige & Sexton found in their research:

"Although Intranets have been adopted widely by large construction companies, their developments have not yet made adequate changes in its nature of content and there is significant potential for improvement. We found that most of the Intranets were populated with static content rather than dynamic content. Hence its capability did not adequately address the capturing of good practices and enabling knowledge sharing within their companies" (2007, p. 419).

The underlying problem with many web-based collaboration tools is that they are digital translations of processes that existed prior to the Web. For example, most intranet and document management systems are typically designed and implemented as centralised and

regulated filing cabinets that provide a current archive of the formal design documentation (Wilkinson, 2005, Chapter 5). Whilst these tools support the management and auditing of project information related to architectural collaboration, their centralisation and regulation restricts the team's ability to inclusively and comprehensively record and reflect upon the design process. In contrast, the Web itself is a successful medium for facilitating and recording conversations, because people can interact with each other in a simple, decentralised and flexible manner (Böhringer & Richter, 2009, p. 293).

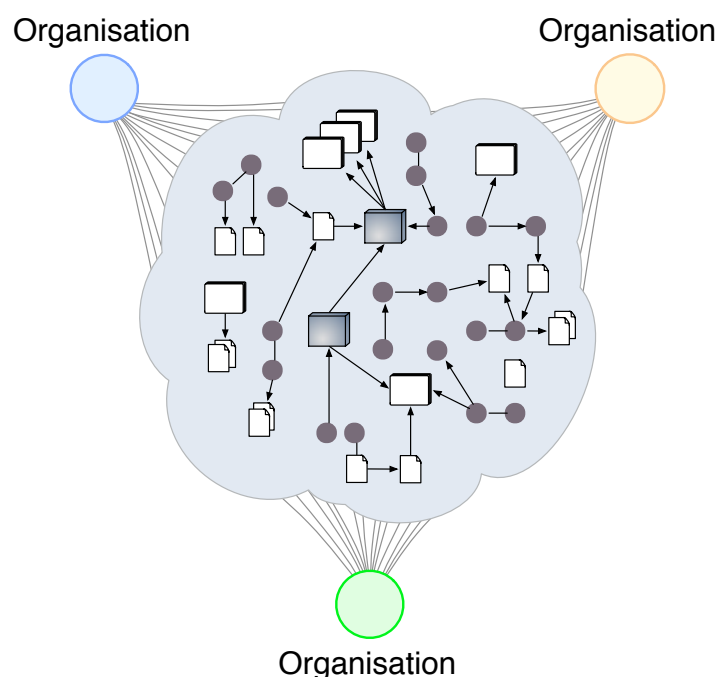
An architectural collaboration system modelled on the Web would not promote the creation of isolated pockets of project information. Instead, the digital messages and related artefacts (digital models, images, etc.) that describe a project team's collaboration interactions would be perceived of as a distributed, interconnected virtual cloud of information. Any member of the team would be capable of contributing to the design process by publishing information to this virtual cloud. To ensure that other members of the team can reliably access this information, the majority of these messages and related data would be published in accessible formats. The emphasis of the information published to this virtual cloud would be on describing and contributing to the project's design process. Like the Web, each digital message or artefact published on the project's information cloud would be assigned a unique Universal Resource Identifier (Berners-Lee, 1996). The assignment of these addresses would allow each contribution to be easily referenced by other resources within the project. Team members could efficiently compile meaningful descriptions of key events, and the project's overall Building Story by using hyperlinks to loosely join the relevant activities and artefacts recorded within the project's information cloud. This process of recording and reflecting upon architectural collaboration interactions in a web-centric manner is called Hyperlinked Practice. Digital collaboration tools that facilitate Hyperlinked Practice stand to improve collaboration within project teams, because participants will be more capable of contributing to, and reflecting upon, a comprehensive digital record of the project's design process.

4.2.2. Adopting Hyperlinked Practice to record Building Stories

Hyperlinked Practice is intended to influence the industry's perception of how information should be recorded and shared within project teams. The current trend of employing isolated, complex and highly regulated Building Information Models and document management systems may allow a project's information to be managed efficiently, but the

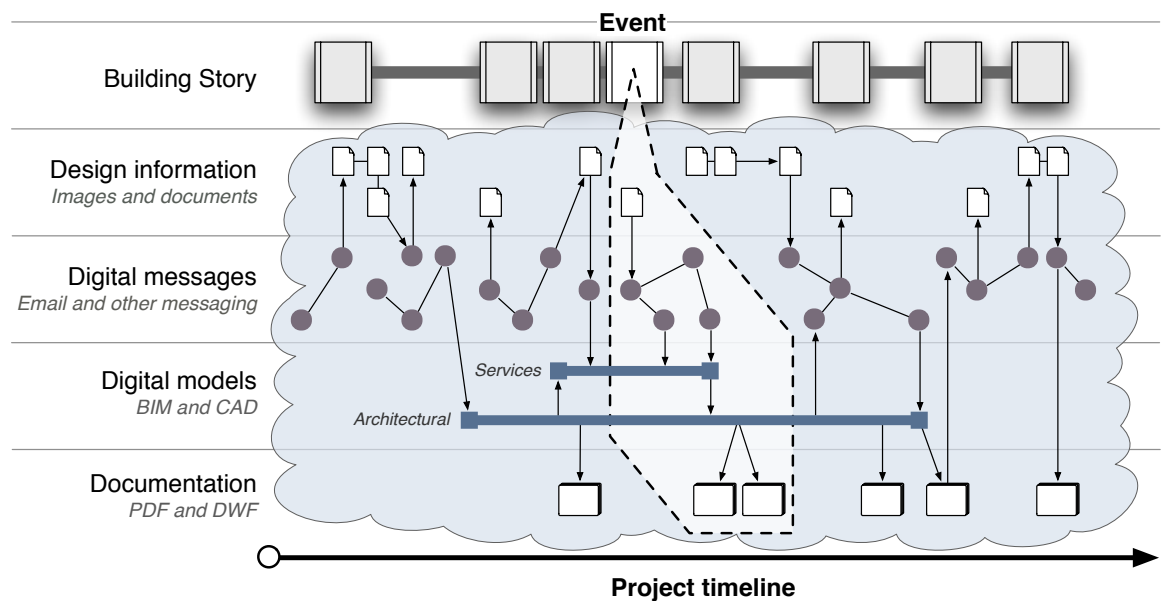
restrictions and participation barriers they impose limit the team's ability to create a comprehensive digital record of the project's design process. Hyperlinked Practice is a more inclusive alternative that promotes creating a virtual project information cloud that encompasses the team and their digital contributions to the design process. The digital resources within an project information cloud will be distributed throughout the team, but from the perspective of those involved it will be presented as a single, common space for collaboration (see Figure 4.2). Like the Web, contributions to a project's information cloud will be unregulated, and any member of the team will be capable of contributing to it. To protect intellectual property and private information, the ability to access data within the information cloud can be restricted by the content owner, or any other authorised members of the project team. Every digital message and resource published to a project's information cloud will be assigned a unique URI, which will enable them to be easily referenced using hyperlinks. It is these links that will help to establish the digital message or resource's place and relevance within the design process. For example, the hyperlinks within a project information cloud can be analysed using the concept of PageRank (Page, Brin, Motwani & Winograd, 1998) to identify significant digital messages and resources. This significance is based on how many hyperlinks within the project information cloud link to a digital message or resource. Thus, if a large number of hyperlinks point to an artefact, it is likely to have played an important role during the design process.

Figure 4.2: The distributed nature of a project's information cloud



Constructing Building Stories within this environment is a case of publishing a document to the project's information cloud that includes a description of the event's characteristics, and hyperlinks to relevant digital messages and resources within the cloud. This process of highlighting digital artefacts within the project's information cloud is illustrated in Figure 4.3. Wikis have been shown to be ideally suited to this task, because they allow the interested parties to collectively craft a comprehensive overview of a design-related event (Burry, Burrow, Amor & Burry, 2005).

Figure 4.3: Constructing a Building Story event within a project's information cloud



4.2.3. The need for fundamental principles of Hyperlinked Practice

A set of fundamental principles are required to guide the creation and deployment of digital collaboration tools that can facilitate Hyperlinked Practice. Establishing this set of principles is important considering the rate of information technology change within the Web and AEC industry.

The principles of Hyperlinked Practice will be derived from the proven principles of the World Wide Web (Berners-Lee, 1998a), but will be directed at addressing the digital collaboration challenges faced by architectural project teams when recording and reflecting upon the design process. Digital collaboration tools that embody these principles will serve to promote and facilitate Hyperlinked Practice today, and in the future.

4.3. Summary

Constructing Building Stories within industry project teams would promote more effective collaboration because participants would be better able to understand design requirements, issues and decisions. Constructing Building Stories requires that the design process be inclusively and comprehensively recorded. Unfortunately, this record is currently inconsistent and difficult to access due to the level of digital fragmentation within project teams, and the difficulties that arise when information is consolidated within complex Building Information Models. To address this problem it is proposed that the AEC industry adopt Hyperlinked Practice, which is the creation of a distributed cloud of interconnected information describing the project's events, activities and digital artefacts. Digital collaboration environments that facilitate Hyperlinked Practice will allow project teams to more efficiently and comprehensively record and reflect upon the design process. This is made possible because digital tools embodying the principles of Hyperlinked Practice will share the same simple, decentralised and flexible characteristics as the Web. The next chapter identifies and describes the fundamental principles of Hyperlinked Practice by analysing how the proven principles of the Web can be applied to the digital collaboration challenges faced within project teams.

5. The Principles of Hyperlinked Practice

Establishing the fundamental qualities of a digital project information cloud

To promote more effective collaboration it is proposed that the architecture, engineering and construction (AEC) industry adopt Hyperlinked Practice, which is the creation of a distributed cloud of interconnected information describing the project's events, activities and digital artefacts. The principles of Hyperlinked Practice have been derived from the proven principles of the World Wide Web, which is the most successful digital medium for enabling large numbers of people to record, share and reflect upon digital information. By analysing the digital collaboration challenges faced within project teams, and how the principles of the Web can help to alleviate them, seven fundamental principles of Hyperlinked Practice have been identified:

- **Situational awareness** - Digital collaboration tools should integrate into the surrounding environment, so that changes that may affect the project are automatically recorded and presented to the team.
- **Ubiquity** - The digital collaboration environment should be based on commonly used processes and technologies, so that any team member may access or contribute to the project's digital record.
- **Comprehension** - All team members should be capable of understanding the purpose, implementation and operation of the project's digital collaboration tools, so that they can appropriately use them in the recording of the design process.
- **Context sensitivity** - Digital collaboration tools should understand and reflect the organisation and current state of the project, so that team members are presented with information that is relevant to the design process and their role within it.
- **Emulative modularity** - The recording and recalling of the design process should not depend on a specific technology or party. Therefore, the digital collaboration environment should be capable of being reproduced or extended by a third-party.
- **Emotive semantics** - The digital collaboration environment should not dictate the semantic system used to record or reflect upon the design process. Instead, the team should be able to define a vocabulary that reflects the uniqueness of each project.
- **Decentralisation** - The digital collaboration environment should respect the team's distributed nature and broad requirements, by not demanding that the design process be recorded in a location that is difficult to access, or controlled by a single party.

5.1. Learning from those who shaped the World Wide Web

The World Wide Web has reshaped the way people access information and communicate with others, by creating a digital environment that anyone can interact with using a web browser running on an Internet-connected computing device. The Web was not the first attempt at such an environment, but it proved to be the most successful because it was unregulated, accessing and navigating web pages required little effort, and the publishing of new content was technically straightforward (Berners-Lee, 2006). From a humble beginning, the Web has now evolved into a rich, highly interconnected ecosystem that touches on all aspects of modern life. This ability to grow and evolve to meet the varied demands of such a broad audience can be attributed to its conceptual foundation. A typical architectural project team represents a microcosm of the Web, because like the Web, those involved have different information needs and technical abilities. Given this similarity, it stands to reason that the underlying concepts that have made the Web successful could be applied to the field of digital collaboration, so that the project team is more capable of recording and reflecting upon the design process. In 1998 Berners-Lee published ‘Web architecture from 50,000 feet’ (1998b) which described the technical design of the Web, and the underlying principles that made it a successful medium for accessing and publishing digital information. The principles outlined by were:

Simplicity - “*A language which uses fewer basic elements to achieve the same power is simpler*” (Berners-Lee, 1998a). Simplicity of both concept and implementation is a key factor for adoption and long term success for any new technology. Adam Bosworth states “that software which is flexible, simple, sloppy, tolerant, and altogether forgiving of human foibles and weaknesses turns out to be actually the most steel cored, able to survive and grow” (Bosworth, 2004). When the Web was in its infancy and its value was questioned, Dougherty, *et al.* highlighted the technology’s simplicity as a factor that would play an influential role in its ultimate success, because “the beauty of HTML is that it’s incredibly simple. As served up by the WorldWideWeb, HTML documents are tremendously straightforward to create. As a consequence, without any of the overhead of the full SGML mindset, HTML documents have multiplied prolifically” (1994, p. 246).

Modular Design - *“If the features can be broken into relatively loosely bound groups of relatively closely bound features, then that division is a good thing”* (Berners-Lee, 1998a). The initial technical specification for the World Wide Web bears only passing resemblance to the Web of today. The principle of modular design enabled this evolution, for whilst “simplicity makes it possible to deploy an initial implementation of a distributed system, extensibility allows us to avoid getting stuck forever with the limitations of what was deployed” (Fielding & Taylor, 2002, p. 116).

Being part of a Modular Design - *“Your own system, no matter how big and wonderful it seems now, should always be designed to be a part of another larger system”* (Berners-Lee, 1998a). The modular and decentralised nature of the Web has made it possible for software developers to quickly and efficiently leverage services and information available on the Web. For example, the Twitter micro-blogging service soon developed a large ecosystem of third-party clients (Yarrow, 2010). The actions of these developers fuelled Twitter’s meteoric growth, and in a relatively short period of time transformed the service into an integral form of communication on the Internet.

Tolerance - *“Be liberal in what you require but conservative in what you do”* (Berners-Lee, 1998a). Within the formal description of the Transmission Control Protocol (TCP) the author Jon Postel defined a Robustness Principle whereby network participants should “be conservative in what you do, be liberal in what you accept from others” (1980). This principle, later known as Postel’s Law become an intrinsic part of the Web when TCP was adopted as its universal, low-level communication standard (TCP/IP). The principle of tolerance is not limited to protocols, but can be applied to the design of the applications themselves. For example, the developers of the web browser Internet Explorer “had this very simple mantra, it does not matter what you give us, we will render it. And every so often we would choke and we would stare at something and say no, no we will render this... Basically the general theory was it takes a licking and it keeps on ticking” (Bosworth, 2005). This tolerance lowered the technical barriers for those contributing to the Web, and ensured a consistent experience for those who were viewing pages that did not fully comply with the HTML standard.

Decentralisation - *“Any single common point which is involved in any operation tends to limit the way the system scales, and produce a single point of complete failure”*

(Berners-Lee, 1998a). A decentralised approach makes the Web capable of accommodating the needs of individual users, and was a pivotal factor in its ability to scale to over 11.5 billion pages. At another level, from the perspective of open source software that drives the majority of the Web, decentralised development is important because “not all developers will necessarily be aligned on all items and tasks. Yet, the processes and tools used should try to promote working towards a common beneficial goal while meeting the individual goals” (Erenkrantz & Taylor, 2003, p. 3).

Test of Independent Invention - *“Avoid conceptual or other centralisation, as not two modules can claim the need to be the unique centre of a larger system”* (Berners-Lee, 1998a). Arguably the most successful and prominent example of this principle on the Web is the hyperlink and the Universal Resource Identifier (URI) which describes the unique Internet address that each page or service has. The ability to directly link to different points on the Web usurps hierarchy and allows independently created websites to integrate with each other seamlessly (Levine, Locke, Searls & Weinberger, 2000, p. 120). The openness of Web standards has ensured that no single party has been able to control or claim overall ownership of the Web, because numerous web server and browser implementations have been able to flourish.

Principle of Least Power - *“The less powerful the language, the more you can do with the data stored in that language”* (Berners-Lee, 1998a). The power of a computer language is often described on a scale of its Turing Completeness which is a measure of how capable a language is of fulfilling the role of Alan Turing’s hypothesised Universal Computing Machine. Languages used to create contemporary software applications are considered Turing Complete, whereas the language of the Web, HTML is purely descriptive and has very little processing logic. According to Berners-Lee, a powerful language is not as useful in a distributed environment like the Web because the information contained within it cannot be easily accessed or reused by other parties (1998a). In contrast, less powerful languages such as HTML, XML and JSON convey information which is capable of being reused in ways the original author could not envision (Wong & Hong, 2007).

5.2. Identifying the principles of Hyperlinked Practice

The principles of Hyperlinked Practice were derived from the proven principles of the World Wide Web. This involved reviewing how the Web's principles, and the technologies associated with them, could address the digital collaboration challenges that restrict a project team's ability to comprehensively record and reflect upon the design process. An overview of these digital collaboration challenges, the applicable Web principles and the principle of Hyperlinked Practice that emerged as a result is presented in Tables 5.1 through to 5.7.

Table 5.1: Deriving the principle of situational awareness

Coping with the influx of digital information within a virtual vacuum	
Digital collaboration challenge	As described Section 3.1.3.i, the adoption of digital collaboration processes has lead to a significant increase in the frequency, quantity and density of information that is exchanged by team members. This poses information management and comprehension challenges, especially for geographically distributed teams. This can lead to misunderstandings, and difficulties when reflecting on the design process.
Applicable World Wide Web principle(s)	<p>Being part of a modular design</p> <p>The Web has demonstrated the almost unlimited potential for information exchange and repurposing. Really Simple Syndication (RSS) is an example of this principle because it enabled people to automatically monitor for changes in weblogs. This development was important because in conjunction with other ‘Web 2.0’ technologies, it heralded a change in the Web’s nature, from a static resource to an interactive and ongoing conversation (O’Reilly, 2005).</p>
Principle of Hyperlinked Practice	<p>Situational awareness</p> <p>Digital collaboration tools should integrate into the surrounding environment, so that changes that may affect the project are automatically recorded and presented to the team. In addition, the digital collaboration tools should expose the digital information it captures in ways that can be consumed by other parties. The outcome of this process will be a digital environment that is more capable of responding to changes in external conditions, and presenting participants with current information that relates to important issues within the team. This responsiveness will in turn serve to improve the team’s understanding of the unfolding design process.</p>

Table 5.2: Deriving the principle of ubiquity

Inconsistent industry adoption and digital fragmentation	
Digital collaboration challenge	As described in Section 3.1.3.ii, the adoption and penetration of different technologies within the industry is inconsistent, which has led to the digital fragmentation of the project team. This has created collaboration barriers and inefficiencies, which in turn restricts some team members from contributing to the digital record of the design process, or participating in conversations.
Applicable World Wide Web principle(s)	Tolerance & the principle of least power The Web has demonstrated that simple formats that any interested party can access and repurpose is valuable within a distributed and unregulated digital environment. The most notable example of this is HTML, which is the information backbone of the Web. Browser applications that render HTML are tolerant of incorrectly formatted documents, and the format is flexible enough to cope with a variety of evolving needs.
Principle of Hyperlinked Practice	Ubiquity The digital collaboration environment should be based on commonly used processes and technologies, so that any team member may access or contribute to the project’s digital record. Undertaking the majority of collaboration interactions using digital processes and formats that are common across the majority of the team will help to heal its digital fragmentation. This will ensure the broadest cross-section of the entire team can contribute to, or reflect upon, the recorded design process.

Table 5.3: Deriving the principle of comprehension

Complexity reduces participation	
Digital collaboration challenge	As described in Section 3.3.1.i, the more complex a digital collaboration technology, the fewer people within the team who will be capable of utilising it. This problem stems from the diversity of the project team, and the digital fragmentation within it. In addition, complex digital collaboration processes pose high adoption barriers, in the form of investments in time and resources. This reduces the likelihood that team members will participate in the digital recording of the design process.
Applicable World Wide Web principle(s)	Simplicity The rapid adoption and growth of the Web has hinged on the ease by which people can interact with, and publish content to it. At a technical level, the Web's underlying simplicity has enabled a vast ecosystem of technologies to flourish with minimal input or direction from the governing bodies.
Principle of Hyperlinked Practice	Comprehension All team members should be capable of understanding the purpose, implementation and operation of the project's core digital collaboration tools, so that they can appropriately use them in the recording of the design process. In addition, the collaboration environment should promote the recording of the design process in a manner that can be understood by the majority of the team. This will ensure it can be meaningfully reflected upon, even if the contributing party is no longer associated with the project.

Table 5.4: Deriving the principle of context sensitivity

A lack of understanding of design requirements and decisions	
Digital collaboration challenge	As described in Section 2.2.1, understanding a project’s design requirements, issues and decisions is important because it promotes more effective collaboration. However, the digital fragmentation of the project team, coupled with the inconsistent and at times inaccurate recording of the design process can make accessing timely and relevant information that relates to these issues difficult.
Applicable World Wide Web principle(s)	<p>Modular design and being part of a modular design</p> <p>Cameron’s Laws of Identity (Cameron, 2005) are a highly regarded set of guidelines for establishing digital identity and trust on the Internet. The Web principles of modular design and being part of a modular design play an important role within these laws, because digital identity cannot be established without integrating with other services and being respectful of the user’s relationships with other systems and organisations.</p>
Principle of Hyperlinked Practice	<p>Context sensitivity</p> <p>Digital collaboration tools should understand and reflect the organisation and current state of the project, so that team members are presented with information that is relevant to the design process and their role within it. The sheer quantity of information within an architectural project means that efficiently reflecting upon the design process requires digital collaboration tools that can automatically filter background noise and highlight important issues. Achieving this requires an understanding of the participant’s role within the project and the relationships they have with other members of the team.</p>

Table 5.5: Deriving the principle of emulative modularity

The continually changing collaboration environment	
Digital collaboration challenge	<p>As described in Section 2.3.4, architectural project teams usually form quickly, and are a unique combination of organisations and professionals. This is a challenging collaboration environment, because each project has specific needs, there is little time to streamline processes, and the composition of the team changes as the project evolves. Consequently, reliably and comprehensively forming a digital record of the design process is difficult, because the project's digital tools are continually changing as the information needs and the composition of the team evolves.</p>
Applicable World Wide Web principle(s)	<p>Modular design & the test of independent invention</p> <p>The Web is formed from a broad range of independently created server and client technologies that implement similar concepts and standard protocols. This diversity has enabled the Web to grow fluidly and to overcome a wide range of challenges, because it is not managed or dependent upon a single party. In addition, this modularity has allowed the underlying client and server technologies to be continually upgraded and replaced without impairing the overall performance, capability or experience of the Web.</p>
Principle of Hyperlinked Practice	<p>Emulative modularity</p> <p>Recording and reflecting upon the design process should not depend on a specific technology or party. Therefore, the digital collaboration environment should be capable of being reproduced or extended by a third-party. This capability will allow digital tools that comprise a project's information cloud to be rapidly deployed, customised, and replaced without damaging the digital record of the design process, or unduly influencing the team's collaboration dynamic.</p>

Table 5.6: Deriving the principle of emotive semantics

Rigid centralisation leads to the filtration of information	
Digital collaboration challenge	As described in Section 3.3.1.ii, creating an accurate digital record of the design process is difficult when restricted to a rigid data-structure or semantic vocabulary, because it places limits on what information can be recorded, and how it can be reflected upon. This poses a problem, because each project is a unique combination of participants, design requirements and issues. As a consequence, when the design process is recorded within a rigid data-structure or semantic vocabulary, it often leads to the filtering and editing of information.
Applicable World Wide Web principle(s)	<p>Tolerance & Simplicity</p> <p>Tagging and the creation of user defined folksonomies has emerged as a popular means of categorising web content (Gruber, 2007). This process embodies the principles of tolerance and simplicity, because instead of integrating a complex data-structure into the fabric of the Web, meaning is manually and externally inferred onto content. The resulting semantic structures are tolerant because they can be manipulated to suit any need.</p>
Principle of Hyperlinked Practice	<p>Emotive semantics</p> <p>The digital collaboration environment should not dictate the semantic system used to record or reflect upon the design process. Instead, the team should be able to define a vocabulary that can categorise content in the simplest and most tolerant means possible. This will enable the design process to be recorded, reviewed and reflected upon in the most appropriate and relevant manner.</p>

Table 5.7: Deriving the principle of decentralisation

Project teams exist within a segmented industry	
Digital collaboration challenge	As described in Section 2.3.3, project teams exist within a segmented and at times combative industry which restricts the recording of the design process if centralised digital collaboration tools are employed. Problems arise because the composition of the team continually changes, participants have different information requirements, and the team is digitally fragmented. Due to this complex environment, the digital record of the design process is usually unevenly distributed throughout the project team in a manner that is difficult to reflect upon.
Applicable World Wide Web principle(s)	Decentralisation The information on the Web is distributed across millions of servers, but from the perspective of the web browser it is a single, cohesive experience. This decentralisation has allowed the Web to rapidly and reliably scale without incurring a performance penalty, and has provided users significant freedom as to how they store, present and experience content.
Principle of Hyperlinked Practice	Decentralisation The digital collaboration environment should respect the team's distributed nature and broad requirements, by not demanding that the design process be recorded in a location that is difficult to access, or controlled by a single party. Decentralisation should extend to the information stored within these systems. A recorded design process that consists of a large number of very small files is ultimately more accessible and flexible compared to a single large file.

5.3. The influence of the principles on the project information cloud

The principles of comprehension, emulative modularity and decentralisation are intended to ensure the project information clouds that form are capable of accommodating the largest and most fragmented of teams. The principle of comprehension aims to ensure that the underlying processes, data formats and technologies that constitute the cloud's digital fabric are understandable and easy to use. This principle will ensure that the broadest possible audience is capable of interacting with this space, and the Building Stories recorded within it. The principle of emulative modularity aims to ensure that undue influence cannot be exerted on the project's information cloud by a single party. To achieve this, any collaboration tool within the project's information cloud should be able of being replaced by a similar, independently developed system. The centralisation of digital information is a key inhibitor to inclusively recording the design process due to the digital fragmentation of the team. To avoid this problem, the principle of decentralisation promotes a project information cloud that is not reliant upon a specific collaboration system or party. Within the virtual space that is formed by a project information cloud, all points of data are considered equal, and any party is free to contribute content without prior permission.

The principles of ubiquity, situational awareness, evolutionary semantics and context sensitively are intended to promote the intelligent distribution of collaboration information throughout the project team. The principle of ubiquity should influence the nature of the digital information exchanged. Rather than stipulating data formats, the emphasis of a project information cloud should be on identifying the most common and useful digital formats that can be used by the majority of the team. As this data is contributed and referenced within the cloud, the principle of situational awareness will ensure that these changes are efficiently brought to the attention of relevant team members. The principle of emotive semantics states that the semantic structures employed by the project team to categorise information related to the design process should be capable of change. This will ensure the recorded design process is an accurate depiction of the project's events and issues. Finally, the principle of context sensitivity promotes a digital collaboration experience where team members are presented with information that is appropriate for their role within the project, or the current design task. This will ensure that participants are able to reflect upon relevant aspects of the design process in a timely manner.

5.4. Summary

Hyperlinked Practice is the creation of a distributed cloud of interconnected information describing the project's events, activities and digital artefacts. Seven fundamental principles of Hyperlinked Practice have been derived from proven principles of the World Wide Web to assist the AEC industry in addressing the digital collaboration challenges faced within project teams. These principles of Hyperlinked Practice are: situational awareness, ubiquity, comprehension, context sensitivity, emulative modularity, emotive semantics and decentralisation.

It is proposed that digital collaboration tools that embody the principles of Hyperlinked Practice will enable project information clouds to form. These clouds will be capable of inclusively and comprehensively recording the design process, and organising it into meaningful Building Stories for reflection and recollection by the entire team.

The next chapter discusses the potential approaches that could be adopted to assess the principles' influence, industry applicability and usefulness in facilitating Hyperlinked Practice.

6. Validating the Principles of Hyperlinked Practice

Determining the best approach for assessing the performance of the principles

The principles of Hyperlinked Practice were identified by analysing how the underlying concepts of the World Wide Web could be applied to assist with the digital collaboration challenges faced within architectural project teams. Assessing the impact of these principles on overall collaboration performance is difficult, due to the complex, multidimensional nature of collaboration. However, an assessment of the principles' influence, industry applicability and usefulness in facilitating Hyperlinked Practice could be used to initially validate these principles. This chapter discusses the benefits and limitations of assessing the principles with a software prototype, comparative case studies or thought experiments and provides the justification for adopting a hybrid software, thought experiment methodology.

6.1. Avoiding the 'better' collaboration quandary

An ideal test of the principles of Hyperlinked Practice would be one that demonstrates whether architectural collaboration is 'better' when project teams use a digital collaboration system that embodies these principles. However, collaboration is a complex, multidimensional concept that is difficult to define and measure. Additionally there are many factors, ranging from the environment through to interpersonal relationships, that could potentially influence collaboration performance (Mattessich & Monsey, 1992). Consequently, even if the same collaboration system is employed, two project teams working independently on a design for the same client, brief and site are unlikely to follow identical communication and decision making pathways. Testing and comparing the overall collaboration performance of different systems is therefore difficult, because external influences cannot be consistently replicated, or controlled, during the test, and therefore the results may not be reliably reproduced (Buckheit & Donoho, 1995, p. 4).

The design outcomes are also not indicative of the collaboration system's performance or success. "We found that designers perceive they have performed better as a team when they agree on, and subsequently follow, a design process, although there was no necessary correlation with the quality of design concept as assessed by independent judges" (MacMillan, Steele, Kirby, Spence & Austin, 2002, p. 176). Likewise, even when design outcomes are similar, each team's opinion on their collaboration system will vary

depending on the external variables which influenced the result. Therefore, even if tests of a collaboration system embodying the principles of Hyperlinked Practice consistently yielded higher quality design outcomes, it cannot be concluded that the principles were instrumental in this result.

Due to the difficulties in directly testing the effect of the principles on overall collaboration, this research focused instead on establishing the initial validity of the Hyperlinked Practice principles by assessing whether they demonstrated their intended influence, were applicable to the AEC industry and were useful in facilitating Hyperlinked Practice.

6.2. Establishing influence, applicability and usefulness

The methodology used to establish the initial validity of the principles of Hyperlinked Practice needs to be able to address the following three questions:

- **Influence:** When the principles are applied within a digital collaboration system, are the hypothesised technical and practical collaboration effects generated?
- **Applicability:** Can the identified principles be applied within the AEC industry without demanding unreasonable changes or collaboration disruptions?
- **Usefulness:** Does a digital collaboration system embodying these principles promote the long-term objective of Hyperlinked Practice?

6.2.1. Influence

Establishing whether the principles can achieve their intended influence is a key initial step in validating these principles. A practical approach would be preferred when testing the influence of the principles, because it would provide tangible evidence that an inferred effect can be achieved. Ideally tests would be within a controlled, preferably repeatable, architectural collaboration situation. Whilst an approach that could quantify the extent of a principle's influence would be beneficial, establishing the necessary control and performance scale would be impractical within the context of this research.

6.2.2. Applicability

The principles need to be applicable to the AEC industry if Hyperlinked Practice is to be achieved. Adoption will not occur if the principles fail to address the identified industry problem (Haymaker, Fischer, Kunz & Suter, 2004, p. 420), or demand impossible changes to existing collaboration processes. In addition, when promoting changes to entrenched systems there is a “demand to prove that it works, and then to answer the question of ‘what if it does not’” (Badger, 1999, p. 3). To satisfy this requirement the approach taken should be able to explore the potential risk and process pitfalls when the principles are applied to industry collaboration situations. An explicit means of proving industry applicability would be to undertake testing within the same environment as the identified problem. However, within the AEC industry this approach is difficult, because of the internal pressures and external forces that shape professional project teams. An alternative strategy would be to simulate relevant industry conditions, or incorporate feedback from industry members.

6.2.3. Usefulness

The adopted approach should be able to explore how useful the principles would be in designing future digital collaboration systems to achieve Hyperlinked Practice, and how the adoption of these systems could influence digital collaboration and the recording of the design process. Whilst theoretical, this process would begin to explore how Hyperlinked Practice could manifest itself, and what role the principles would play.

6.3. A review of alternative approaches

Three relevant approaches were identified as candidates for testing the influence, applicability and usefulness of each principle. These were to develop and test a software prototype, undertake comparative historical case studies, or a series of software thought experiments. A review of these alternative testing strategies, detailed in Table 6.1, indicated that none of the approaches were capable of adequately assessing all three identified aspects of validity.

Table 6.1: An overview of the characteristics of each potential testing methodology

Methodology	Benefits	Shortcomings
<p>Software Prototype</p> <p>Develop and test a digital collaboration prototype that embodies each of the principles. This tool would be tested within a controlled environment, with analysis of usage statistics and user feedback to assess the effect of each principle on its design and operation.</p>	<ul style="list-style-type: none"> • A prototype would provide tangible results from a controlled, practical test. • Conclusions on the validity of principles would be backed by quantitative data. • Each principle would be applied directly to the design and deployment of the prototype. 	<ul style="list-style-type: none"> • A prototype is a significant time commitment and has a high risk of test failure. • The test would constitute a single study of a digital tool operating within a controlled environment.
<p>Comparative Case Studies</p> <p>Select several recent local architectural projects to use as collaboration case studies. Develop a comprehensive profile of each project’s digital processes in order to compare how the principles were directly, or indirectly applied, and subsequent collaboration effects.</p>	<ul style="list-style-type: none"> • Case studies represent actual instances of industry collaboration. This lends credibility to conclusions on applicability. • The digital collaboration processes of projects that span multiple years can be studied, leading to conclusions about the effects of each principle over time. 	<ul style="list-style-type: none"> • Given industry trends, the likelihood of a case study reasonably testing all seven principles is low. • Selecting comparable case studies is difficult given no two collaborations are identical. • Rather than embodied, each principle would be retrospectively inferred from historical accounts.
<p>Thought Experiments</p> <p>Develop a series of software thought experiments that describe in detail how a digital collaboration system embodying the principles would operate.</p>	<ul style="list-style-type: none"> • Allows the principles to be comprehensively applied to complex, yet to be built, digital collaboration systems. • Less risk and time commitment when compared to developing a prototype or undertaking comprehensive case studies. 	<ul style="list-style-type: none"> • The application and evaluation of a principle’s effects would be theoretical and not supported by practical evidence.

6.3.1. Develop and test a software prototype

A software prototype embodying the principles of Hyperlinked Practice is the most tangible means of evaluating their influence on a collaboration tool's design and operation. The justification for prototype development is that no existing architectural collaboration system embodies all seven principles (see Table 6.2). Validating the principles using existing software would require either several software tools to be concurrently tested within a single environment, or individual tests of each principle using different software tools in different locations. In both cases the design, deployment, monitoring and analysis of the principles' influence would be complicated by the need for multiple software tools and testing processes. Utilising existing software also risks results being influenced by a participant's previous experience with, or bias towards, an existing digital tool. In contrast, developing a software prototype ensures all participants begin the test with similar knowledge of the digital tool and its architectural collaboration potential.

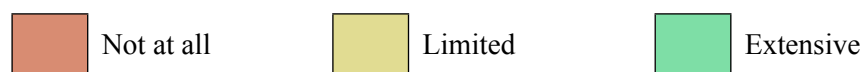
The value of a working prototype is that many of the technical and collaboration complexities inherent within 'real world' situations can be discovered and worked through during the process of designing and executing the test. Tests that do not directly engage in a collaboration process risk making optimistic, or incorrect, assumptions on operational conditions, or the participant's behaviour. A software prototype could also generate a considerable amount of usage statistics, which can be leveraged to gauge the influence of the principles. This quantitative data could be combined and compared with qualitative results from user feedback to form a more comprehensive understanding of how a principle's collaboration influence was demonstrated.

The greatest risk with a software prototype is that a successful test requires a considerable investment in time and resources. This investment is needed to ensure that the collaboration effects measured are robust and ultimately applicable to the industry. To ensure the test reflects a typical architectural project, the prototype must be frequently used by the same group of people in a project spanning many months. This poses a significant risk as inconclusive results may require aspects of the test to be repeated, or remain unanswered. Given the resources available, and the test's duration and complexity, it would be impractical to develop multiple prototypes, or repeat the test under different conditions. As a consequence, the results available to draw conclusions from would be limited to a single digital tool, applied within a specific collaboration environment.

Table 6.2: The embodiment of the principles of Hyperlinked Practice within existing digital collaboration technologies

Digital Collaboration Technology		Comprehension	Modularity	Decentralisation	Ubiquity	Situational Awareness	Context Sensitivity	Dynamic Semantics
Message Driven Conversation	Email Correspondence	Extensive	Extensive	Limited	Extensive	Not at all	Limited	Limited
	Mailing Lists and Discussion Forums	Extensive	Extensive	Not at all	Extensive	Not at all	Limited	Not at all
	Instant Messaging Services	Extensive	Limited	Limited	Extensive	Not at all	Not at all	Not at all
	Internet Voice and Video Conferencing	Limited	Limited	Limited	Limited	Limited	Not at all	Not at all
Project Collaboration Systems	Simple File Storage and Sharing	Extensive	Extensive	Extensive	Extensive	Not at all	Not at all	Limited
	Organic Company Intranet	Limited	Extensive	Not at all	Limited	Limited	Limited	Extensive
	Application Integrated Web Services	Not at all	Limited	Limited	Not at all	Extensive	Extensive	Limited
	Third-party Project Websites	Extensive	Limited	Not at all	Limited	Limited	Limited	Limited
Collaborative CAAD Workspaces	Linked Independent Files	Extensive	Limited	Extensive	Extensive	Not at all	Limited	Limited
	Master Model with Satellite Copies	Not at all	Not at all	Limited	Not at all	Not at all	Limited	Not at all
	CAAD Server with Remote Clients	Not at all	Limited	Limited	Not at all	Extensive	Limited	Limited
	Hybrid Approaches	<i>Depends on the combined C-CAAD technologies</i>						
BIM	Standalone BIM	Not at all	Not at all	Not at all	Not at all	Limited	Extensive	Not at all
	BIM Server with Remote Clients	Not at all	Limited	Limited	Not at all	Extensive	Extensive	Not at all

Embodiment of the principle within the digital collaboration technology:



6.3.2. Comparative case studies of digital collaboration

A comparative case study methodology could use historical examples of industry digital collaboration as the basis for analysing the influence and applicability of the principles. As illustrated in Table 6.2, the principles of Hyperlinked Practice are evident within existing software, but they are not evenly distributed. By identifying when and how the software tools used in previous projects embodied the principles, their practical and technical influence could be assessed. Case studies are retrospective, which limits their ability to test how useful a principle is in achieving Hyperlinked Practice. However, different case studies do provide an opportunity to compare and contrast examples where the principles were applied to different degrees, and their subsequent influence on their intended effects.

The benefit of a case study methodology is that the case studies can be based within the industry, which enables the industry applicability of the principles to be thoroughly tested. Undertaking case studies would highlight digital collaboration issues unique to professional architectural collaboration, that cannot be simulated in external testing environments. For example, staff turnover is a significant collaboration confounder because this often leads to knowledge loss, unexpected problems, and changes in the team dynamic. Accurately simulating such events in an environment that is not industry based is difficult. Another benefit of historical case studies is that they enable long-term digital collaboration processes to be analysed within a relatively short period of time. From the perspective of validating the principles this is important, because it provides insight into the effect of time on a principle's collaboration influence and applicability to the industry problem. Theoretical work or practical tests cannot reasonably simulate this, because time's impact on the principles' collaboration effects is not yet understood. For example, the inferred influence and applicability of a principle may hold true for one phase of a project, but maybe different in another phase, or after a period of time has elapsed.

The difficulty with this methodology is that the likelihood of establishing substantial, direct evidence of the principles' effects within the cases studied is low. In addition, accurately comparing different projects, for example a three bedroom house to a ten storey industrial complex, could be difficult due to these projects having different external influences. Using projects of similar scope could partially address this issue, but even then there are likely to be a number of different external influences acting on them.

A final shortcoming of the case studies approach is that determining a principle's influence relies on historical records and the recollection of events by participants. This is problematic because historical digital records may be incomplete and a participant's memory is often unclear or biased. In contrast, the test data generated by a software prototype or discussion group is immediately recorded, leaving less scope for missing data, inconsistencies or recall bias within the results.

6.3.3. Thought experiments

The thought experiment approach could develop and evaluate theoretical situations where the principles are comprehensively embodied within the design of a hypothetical digital architectural collaboration environment. These hypothetical designs could allow the principles to extensively influence the collaboration system's design well beyond what is practical within a software prototype. The hypothetical influence and applicability of each principle within a software prototype could be discussed. Compared to software prototypes and case studies, a thought experiment is a very efficient means of identifying the usefulness of the principles in facilitating Hyperlinked Practice.

Similar to software prototyping, a thought experiment methodology would allow all seven principles to be embodied and evaluated within a single collaboration system. This would allow the overarching concept of Hyperlinked Practice to be explored, and in the process demonstrate how the principles could facilitate it. When compared to a software prototype, thought experiments pose less risk, because a working system does not have to be developed or deployed prior to the receipt of meaningful feedback. This promotes an iterative testing approach where a series of digital collaboration thought experiments can be proposed and evaluated within the time it takes to complete a single software prototype. The outcome would be a well developed, comprehensive design for a new digital collaboration system that was driven completely by the principles rather than the constraints of historical precedent or practical necessity. Ultimately this would benefit the validation process because the principles' intended effects, and potential benefits or shortcomings, would be more apparent.

A thought experiment methodology would promote future-looking discussion and serve to develop a long-term understanding of Hyperlinked Practice. The risk of a thought experiment is that it lacks practical grounding, and as a result the system proposed or opinions expressed maybe incorrect or unrealisable. In addition, thought experiments

cannot take into account the process uncertainties and external variables that affect architectural collaboration. Whilst some events can be designed into the problem scenarios, it is more difficult to postulate a project team’s response, and how this could influence digital collaboration and the principles’ effects.

6.4. A hybrid software prototype/thought experiment testing strategy

A hybrid software prototype / thought experiment testing methodology was identified as the best means of assessing the initial validity of the principles of Hyperlinked Practice. The software prototype, tested within industry-like conditions, would be able to demonstrate whether the principles demonstrated their intended influence in practice. Following this, the series of thought experiments would be able to explore the usefulness of the principles in facilitating Hyperlinked Practice.

A hybrid approach was adopted because a single methodology could not adequately assess the influence, applicability and usefulness of the principles. Applying all three methodologies would have comprehensively covered all these aspects of validity, but this exceeded the time and resources available for the research. Table 6.3 provides an overview of the possible methodology combinations. The comparison illustrates that even when two methodologies were applied, all three aspects of validity were not comprehensively tested. However, all three aspects were addressed to some degree, and when drawing conclusions the results from the two strongest aspects could reinforce the weaker area.

Table 6.3: Ability of the methodology combination to test the validity of the principles

Methodology Combination	Influence	Applicability	Usefulness
Prototype / Case Studies	Comprehensively	Very capable	To a limited degree
Prototype / Thought Experiments	Very capable	To a limited degree	Comprehensively
Case Studies / Thought Experiments	To a limited degree	Comprehensively	Very capable

Ability of the methodology combination to test the aspect of validity:

To a limited degree
 Very capable
 Comprehensively

A methodology that was focused on testing influence and usefulness was preferred because the principles are of most value to the industry if they influence the projects as expected, and usefully work towards the long-term objective of Hyperlinked Practice. Testing for industry applicability is important, but given the complexities of testing within this environment and the resources available, it was more important to demonstrate that the principles can facilitate Hyperlinked Practice and influence the recording of the design process. The software prototype / thought experiment methodology was the most balanced means of testing the principles' influence and usefulness in facilitating Hyperlinked Practice and can be used to infer applicability to a limited degree.

6.5. Summary

A hybrid software prototype / thought experiment methodology is the most appropriate means of validating the principles of Hyperlinked Practice within the context of this research. Due to the limited time and resources available for this research, these tests would be an initial validation of the principles, and would not serve as a definitive measure of their performance. The emphasis of these validation tests is to demonstrate the principles' influence, industry applicability and usefulness in facilitating the long-term objective of Hyperlinked Practice. The next chapter describes the design of the software prototype and the associated tests for demonstrating each principle's inferred collaboration influence. To enhance industry applicability, this process emphasises the identification and simulation of industry collaboration conditions within a controlled environment.

7. Testing the Principles within a Software Prototype

Designing Reasonate to evaluate Hyperlinked Practice within a digital design course

A software prototype named Reasonate was developed and tested to demonstrate the influence of Hyperlinked Practice and its principles. Testing the principles was necessary to provide evidence that they are able to record the Building Story and benefit collaboration. For a principle to be beneficial, it needs to demonstrate the following properties:

- **Influence** - The desired collaboration effect is achieved when the design or implementation of a digital tool is consciously influenced by the principle.
- **Applicability** - The technical and collaboration effects of the principles are applicable within the AEC industry and the identified collaboration problem.
- **Usefulness** - When applied to digital collaboration tools used within the AEC industry, the principle and its associated effects facilitates Hyperlinked Practice.

Given the time and resources available, the most appropriate means of testing whether the principles demonstrate these properties was to apply them to the design of a software prototype, and to explore their long-term potential using a series of thought experiments.

The Reasonate software prototype was tested within the BBS303 'Digital Craft' course at Victoria University of Wellington. This course was used as a testing ground because it simulated relevant industry conditions, and was insulated from many of the external forces and complexities that confound research inside a professional project team. The principles of Hyperlinked Practice were used to derive many of Reasonate's functional traits. Tests were then identified that could establish whether this functionality was able to influence collaboration in the manner predicted by the principles. Combined, these identified traits and tests informed the design and implementation of Reasonate, which was developed as a purpose-built architectural blogging system. When the testing process was nearing completion, the students involved were asked to complete a questionnaire that reviewed their Reasonate usage patterns, and its perceived influence on the collaboration process. These personal reflections were used in conjunction with analytical data collected during testing to evaluate the influence of the principles, and the ability of Hyperlinked Practice to record the Building Story.

7.1. The testing environment for the Reasonate prototype

The prototype was named Reasonate because it reflected the underlying research objective of communicating design intent throughout a project team. Reasonate was tested within Victoria University's BBS303 'Digital Craft' course, because compared to a professional project team it posed less practical hurdles or external confounding factors. Test results were still generalisable to the broader industry because both environments shared key collaboration similarities. Reasonate played a central role during the course, as students used it to document their collaboration processes and exchange digital information.

7.1.1. Justification of the test environment

Testing Reasonate within a professional project team would be a robust demonstration of the principles' industry applicability, but this was impractical for the following reasons:

- **Trust** - Architectural projects are intense and high-risk environments, where the majority of those involved make conservative technology and process decisions (Egan, 1998, p. 37). Convincing professionals to adopt and consistently use an unproven prototype within such an environment is a difficult proposition.
- **Duration** - The time taken to complete a typical architecture project was too long to be used as the prototype's testing duration. Testing the prototype during a specific phase of the project, or between two arbitrary dates, was also problematic given the inconsistent nature of collaboration activity (Laepfle, Clayton, Johnson & Parshall, 2005, p. 110). This inconsistency could have led to a situation where too few results were recorded to be useful when analysing the principles' effects.
- **External factors** - Architecture projects are influenced by a range of external factors that can force changes on the scope of the project, its timeline and the composition of the team (Barrett & Stanley, 1999, p. 14). Many of these factors could complicate the testing process and cast doubt on the recorded results. A risk also exists that the project itself could be cancelled, which would lead to the premature end of testing.
- **Support** - A prototype used within a professional project team would demand a level of support that is impractical given the time and resources available for the research. Additionally, any meaningful commitment by a professional team would require the prototype remaining available and reliable once testing was complete.

In contrast to a professional architecture project, a university-based digital design course does not pose the same testing concerns:

- **Trust** - Universities are a place of research, and the majority of staff and students are open to using new technologies and processes if they maybe of benefit to them.
- **Duration** - Most university courses last a semester, which is approximately four months. The workload during this time is tightly regulated by course coordinators. This short timeframe allowed the prototype to be tested for the duration of the course, and the workload during this time was consistent.
- **External factors** - Unlike an architectural project, few external forces act on a university course. Course requirements and student registrations are confirmed at the beginning of the semester, and there is an extremely low probably of the course being cancelled prior to, or during testing.
- **Support** - University courses have coordinators and tutors that can be called upon to provide additional support for the prototype. Additionally, the lower financial and process risks associated with student work, compared to that of a professional team, would not demand the same level of support or reliability.

The collaboration challenges within a university-based digital design course are not identical to those found in a professional architecture team, but for the purposes of the research they are comparable. A breakdown of some of the similarities and differences between these two environments are listed in Table 7.1.

The two environments are comparable because both are comprised of opinionated people who are working together over a relatively long period of time to solve a design problem. Unlike a university course, a professional project team is comprised of people from a broad range of professions, who are dealing with complex financial pressures. Inter-disciplinary disagreements and financial compromises are important events, but these are collaboration situations that are best studied independently in future research.

Table 7.1: Collaboration similarities and differences between a university testing environment and a professional project team

Similarities	Differences
<p>Time constraints - Both environments have strict deadlines and time pressures.</p> <p>Duration - Both environments feature design problems that span a relatively long period of time.</p> <p>Content - The problems that are being addressed are inherently related to the field of architectural collaboration.</p> <p>Structure - The teams in both environments report to a ‘client’, and are relatively egalitarian in nature. Within a course, the ‘client’ is the coordinator.</p> <p>Opinionated - The team members in each environment are generally willing and able to express conflicting opinions.</p>	<p>Diversity - Professional project teams involve a variety of professions, whereas a university course is usually limited to a single profession.</p> <p>Complexity - The problem solving process within a professional project team is more complex and involves greater compromise, often because of financial considerations.</p> <p>Experience - Industry professionals typically have a lot of experience in working as part of a team. In contrast, university students generally undertake projects independently.</p>

Victoria University’s BBS303 ‘Digital Craft’ course was an appropriate test environment, because it involved a relatively small number of students working in groups to undertake computer-centric project activities. The course, coordinated by Michael Donn, was well established within the Architecture School at Victoria University. Given this history, a strong understanding of the course’s requirements, workload and outcomes was leveraged during the design and deployment of the prototype. Additionally, students who took the course did so because they wanted to gain a better understanding of how computers could be used in the architectural design process. This inferred they would not be against using the prototype, especially if it was shown to be of value to their coursework.

Compared to many other architecture courses at Victoria University, the design component within BBS303 ‘Digital Craft’ was relatively small due to its emphasis on computing. However, at the time it was one of the few papers offered by the School of Architecture with an established record of combining digital modelling, teamwork and process documentation into the majority of its coursework. Consequently, testing Reasonate within this context was more appropriate because at the time the design-centric courses lacked many of these other essential characteristics.

7.1.2. Composition of the test audience

The BBS303 ‘Digital Craft’ course of 2006 had sixty registered undergraduate students, eight tutors and one coordinator. The undergraduate students involved were in their third or higher year of study towards a degree in architecture or building science. By this time of their tertiary education, all of the students had a moderate understanding of how to use computers, email, the Web and standard office productivity software tools.

The eight tutors were all undergraduate students who had completed the course in previous years. Tutors were not expected to actively use Reasonate, but they were introduced to the prototype. In partnership with the course coordinator, they acted as the first line of support for any Reasonate issues.

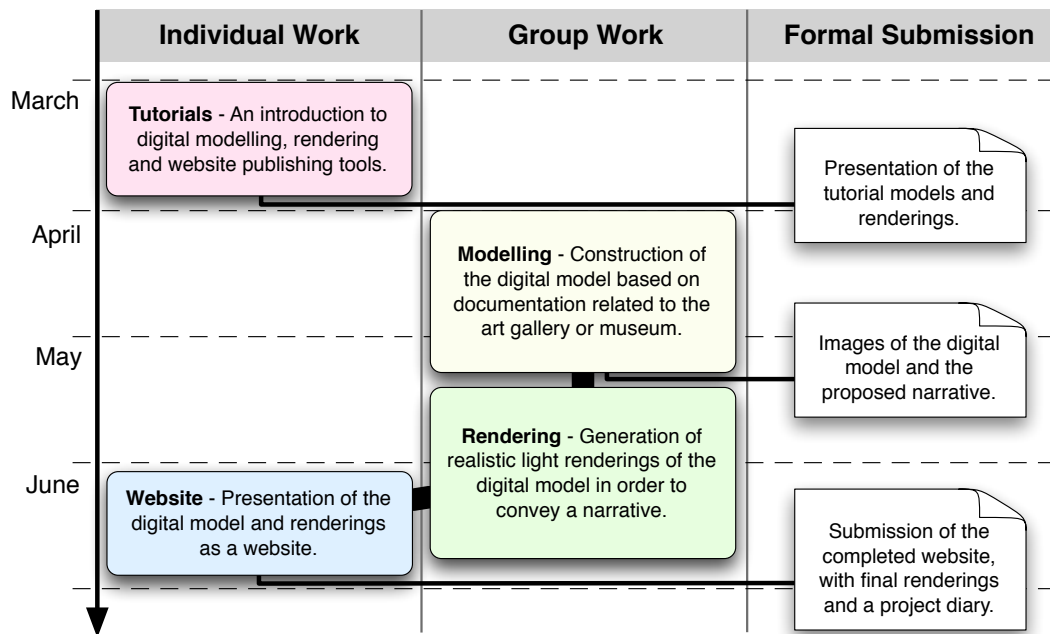
7.1.3. Course objectives and outline

The primary objective of the BBS303 ‘Digital Craft’ course was to introduce students to the various roles computers could play in the architectural design and collaboration process. The practical coursework focused on three digital tasks that are prominent within architectural design and communication:

- **Digital modelling** - The collaborative construction of complex, 3-dimensional digital model using CAD and BIM software packages.
- **Lighting simulation** - The generation of realistic lighting simulations of digital models using ray-trace and radiosity-based light rendering software.
- **Web communication** - The communication of architectural information on the Web through the publishing of simple websites and online media.

For the majority of the course, students worked in small teams of two or three people. Their primary task during this time was to accurately and comprehensively model a notable, real-world art gallery or museum. Once modelled, the student groups prepared digital renderings of the model in order to convey an identified architectural narrative. Finally, each student presented these renderings, the digital model and a project diary summarising their group’s actions within a personal website. Figure 7.1 illustrates an outline of this coursework, its timeline, and the formal submissions during this process.

Figure 7.1: An outline of the BBS303 'Digital Craft' course



7.1.4. Collaboration within the course

The majority of the coursework within BBS303 'Digital Craft' was conducted in small groups. Within these groups, students decided on the most appropriate digital tools and strategy for modelling their chosen art gallery or museum. Often the quality and quantity of published architectural drawings for these buildings was minimal, so the students had to collectively work through design problems that ranged from understanding the architecture's fundamental concepts, through to the specifics of important construction details. Once this task was complete, the students identified an architectural narrative that they could convey through simulated light renderings of the digital model. To ensure that these renderings were consistent, students were expected to clearly document the properties of the materials and light fixtures used. Throughout this process, the students were required to document their collaboration workflow, the problems experienced, and subsequent decisions made. This diary of project activity was included within the personal website that formed their final course submission.

The majority of the students in BBS303 'Digital Craft' were based in the same building, and shared common workspaces. As a result, the majority of collaboration interactions during the course were expected to occur in person. Students were encouraged to use Reasonate to record the outcomes of these conversations, and to support the exchange and archiving of digital files related to them.

7.1.5. The role of Reasonate

Reasonate's primary role was to help students to collaboratively record the digital processes they were undertaking and to assist in the exchange of digital information between peers. In previous years these tasks were manually undertaken by students, so in this sense Reasonate was supporting their workload, not adding to it. This factor was important, because adoption and use was less likely to be affected by the perception of Reasonate as an additional burden.

To satisfy the requirements of the BBS303 'Digital Craft' course, Reasonate had to support the following functionality:

- **Media sharing and archiving** - The uploading and presenting of digital media, such as CAD or BIM models, images and related documents.
- **Diary** - A record of the student's output and the processes followed by their team.
- **Discussion** - Conversations between students, tutors and coordinators that related to their project activities, and the digital models and images that were generated.
- **Teams** - The grouping of students into small project teams that were named after the art gallery or museum they were modelling.
- **Progress** - A simple means for students, tutors and coordinators to review the progress of individuals and teams.
- **Student websites** - The publishing of the student developed websites to an Internet accessible web server.

The majority of the students did not have computers of their own, and instead relied upon the computing resources at the School of Architecture. No additional software could be installed onto these computers, which meant that if students were to reliably access Reasonate, it had to be via a web browser.

7.1.6. Ethics approval

Ethics approval for the testing of the prototype within the BBS303 'Digital Craft' course was granted by the Victoria University of Wellington Ethics Committee in May 2006.

7.2. Testing the influence of each principle

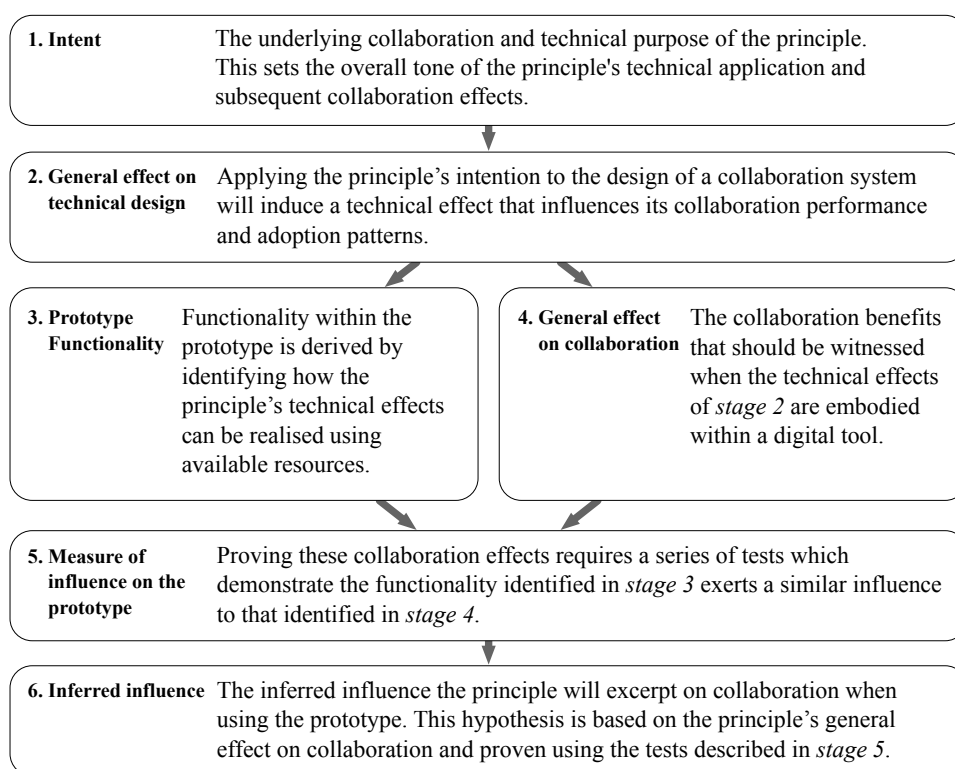
The process used to develop the prototype's functionality was divided into six steps, which are described in Figure 7.2. Following this process helped to identify:

Functionality - Functional characteristics of the prototype that were derived by applying the principles to its design.

Collaboration effect - The effect on collaboration interactions between participants when using a prototype that embodies the principles.

Tests - A measurable indication of each principle's collaboration effect.

Figure 7.2: The process of developing practical tests & hypothesis for each principle



This systematic process was applied to help ensure that the tests developed to demonstrate the principles' influence satisfied the following criteria:

Abstraction - The measured effects could be attributed to the underlying principles, rather than the prototype's specific design or underlying technologies.

Reproduction - A third-party could achieve similar results using a different prototype that used the same process to derive its functionality.

Repeatability - Similar results could be achieved if testing of the prototype was repeated within a different architectural collaboration environment.

7.2.1. Testing the principle of situational awareness

Principle:	Situational Awareness
Principle's intent:	A collaboration system should assume it is part of a larger information ecosystem. To best serve its users the system should strive to integrate into this environment as tightly as possible.
General effect on technical design:	The technical implementation should integrate with third-party systems by sharing and consuming data streams. This will provide more timely and integrated access to collaboration information (Christiansson 1998, p. 204). An efficient means of reviewing and searching this content should also be provided.
Prototype functionality:	The prototype generated a range of RSS feeds that could be subscribed to by external applications. Email notifications were also available so that team members could be immediately alerted of new content. Finally a search engine was integrated into the prototype so that participants could locate historical information, or monitor new content for specific terms.
General effect on collaboration:	It is inefficient and impractical for team members to monitor all collaboration activity. Delivering relevant, aggregated information to their preferred application or device ensures participants stay informed of events, thus improving their comprehension of the project (O'Brien et al 2002, p. 102).
Measure of influence in the prototype:	Situational awareness was improved if participants regularly published content and leveraged the prototype's toolset to stay informed of project activity. To measure this content growth throughout the test's duration was monitored to establish whether regular contributions from participants occurred. A questionnaire at the end of the testing process gauged the utilisation and perceived value of the prototype's monitoring functions (RSS, email and search) amongst participants.
Inferred influence:	A collaboration system embodying the principle of situational awareness will actively acquire collaboration information from users and external sources. Instead of checking for new data, users will take advantage of search and syndication mechanisms that enable them to passively scan for significant events.

7.2.2. Testing the principle of ubiquity

Principle:	Ubiquity
Principle's intent:	The technologies that enable digital collaboration should be present within the broadest practical audience. This availability minimises potential barriers to adoption and participation.
General effect on technical design:	The technical implementation should leverage widely accepted technology concepts and standards in its design and implementation. This may restrict functionality, but it will ensure all team members can interact on an equal footing using widely available tools (Varkonyi 2009, <i>see</i> 'Diverse world').
Prototype functionality:	The prototype was Web-based and accessible through any standards compliant Internet browser. Whilst any content could be published, standard formats with viewing support within the browser (e.g. images) provided users with a more interactive experience. Published content was syndicated via RSS feeds so that content could be monitored by various reader applications.
General effect on collaboration:	The widespread and easy availability of a system encourages consistent use by all members of the project team. This improves collaboration by reducing participation barriers (McCall 1999, p. 72) and ensures that the contributed information remains accessible for the duration of the project.
Measure of influence in the prototype:	Usage patterns and the content types submitted to the system were the best indication of ubiquity's influence. Prototype use was monitored to determine at what times during the day it was used. To support this data, participants were asked when and where they used the prototype in a questionnaire issued at the end of the test. Finally the file types published to prototype were analysed to identify whether a preference existed for standard, Internet browser supported bitmap files.
Inferred influence:	A collaboration system embodying the principle of ubiquity will be used more frequently by the entire team throughout at project. This emphasis will encourage participants to publish the majority of their content using standard formats instead of relying on more powerful, but less accessible forms of data.

7.2.3. Testing the principle of comprehension

Principle:	Comprehension
Principle's intent:	The concepts, interface and underlying technologies of a collaboration tool should be easily understood by participants and those interested in its technical implementation.
General effect on technical design:	The technologies and design patterns used in the technical implementation must be "the simplest thing that could possibly work" (Beck 2001, p.113). This emphasis leads to software outcomes with a high probability of being understood, adopted by and built upon by informed third-parties.
Prototype functionality:	The prototype had a concise feature-set that was presented in an uncluttered manner. To ensure a smooth learning curve, functionality was incrementally introduced throughout the duration of the test. The prototype was Web-based and accessed through browser applications that were widely available and understood by the majority of test participants.
General effect on collaboration:	A tool that is easy to comprehend will have a low barrier to entry and thus will see broad and regular use within a project team. Encouraging participation increases communication within the team and therefore the value of the underlying collaboration system (Hendler & Golbeck 2008, p.15).
Measure of influence in the prototype:	Adoption and usage patterns irrespective of a participants' prior technical experience are the best indication of comprehension. Therefore if adoption rates are consistently high the principle positively influenced the prototype's collaboration performance. Use of the prototype was monitored throughout the duration of the test to establish rates of adoption. Participants' prior experience and perceived usage patterns was assessed by questionnaire on completion of the test.
Inferred influence:	A prototype embodying the principle of comprehension will be rapidly adopted by participants, irrespective of their previous technical experience. Usage of the prototype will remain constant throughout the duration of the test as long as it is reliable and clearly presents collaboration exchanges.

7.2.4. Testing the principle of context sensitivity

Principle:	Context Sensitivity
Principle's intent:	Information should be presented in a manner relevant to the collaboration situation. For this to occur, the system must comprehend the state of the project and the team's dynamic.
General effect on technical design:	The technical implementation should capture and convey supporting information about the project and the team's composition. This meta-data can be leveraged to improve the comprehension of participants (Chrabin et al 2004, p. 166) by filtering and highlighting relevant collaboration exchanges.
Prototype functionality:	The prototype was designed from the outset to comprehend team structures and roles so that participants could easily view, navigate and search for content generated by their team. When viewing content, users were presented with associated tags and a "cloud" representation which illustrated how these concepts related to broader issues faced by fellow participants.
General effect on collaboration:	Intelligently filtering content based on a team's composition and contextual meta-data improves the individual's ability to comprehend different aspects of a project relative to its overall goals (Chrabin et al 2004, p. 265). This understanding supports more productive collaboration exchanges with team members.
Measure of influence in the prototype:	The frequency of contributions and their relationship to external events indicates the influence of context sensitivity within the prototype. The growth rates of published content and meta-data was tracked to determine the regularity of contributions and their relationship to significant external events. A questionnaire undertaken at the end of the testing period measured each participant's opinion the act of recording and consuming this content played in their comprehension process.
Inferred influence:	A prototype embodying the principle of context sensitivity will display information to users in a manner that is relevant to the task they are currently performing. This tailored presentation will instill a better understanding of the issues faced by the project team and provide a more efficient digital workspace.

7.2.5. Testing the principle of emulative modularity

Principle:	Emulative Modularity
Principle's intent:	The collaboration system should be capable of being extended or replicated by a third-party without impeding the flow of information between participants or the message's intent.
General effect on technical design:	Adding or removing functionality from the technical implementation should be possible without impacting its reliability or cohesiveness (Papamichael et al 2000, p. 22). This same technical architecture should also allow third-parties to replicate aspects of the system in a compatible manner.
Prototype functionality:	The prototype leveraged existing Web standards such as HTML to create functional components joined together by hyperlinks. This allowed the system to be incrementally built-out during the testing process without impacting the experience of participants. This design pattern enabled an informed third party to independently replicate the functionality of the prototype.
General effect on collaboration:	A modular system ensures consistency during times of technical change, for example the introduction of new functionality or a change in software (Raymond 2003, see ' <i>Rule of Modularity</i> '). This consistency promotes confidence that collaboration interactions will reliably occur during the project.
Measure of influence in the prototype:	The prototype was replicated by a third-party to determine whether modularity ensured a consistent environment. This replica was applied to the same collaboration problem to test whether both systems experienced similar adoption and utilisation patterns. Usage was compared by measuring the rate of published content over time. The length of messages and the types of files published to each system was compared to identify whether the format of content exchanged was similar.
Inferred influence:	A prototype embodying the principle of modularity will be reproducible by a third-party in a manner compatible with the overall collaboration dynamic. However, variances in implementation will influence the behavior of participants, resulting in similar, but not identical usage patterns.

7.2.6. Testing the principle of emotive semantics

Principle:	Emotive Semantics
Principle's intent:	Participants should be free to manipulate the semantic vocabulary used to categorise collaboration content so that the evolving issues of a project can be appropriately recognised.
General effect on technical design:	The technical implementation should apply categorisation as a layer distinct from the source collaboration content. This stratification will provide more flexibility in the way content is described (Sinclair, & Cardew-Hall 2008, p. 16), thus allowing the project's vocabulary to be reshaped over time.
Prototype functionality:	The prototype had a tagging mechanism that allowed teams to categorise content in a manner that suited their explicit needs. Users of the prototype were able to easily reclassify content, or add new layers of categorisation, during the test. To promote comprehension and navigation, the semantic structure was presented to users in a variety of ways, including tag clouds.
General effect on collaboration:	A semantic structure tailored to the needs of the project team will help describe and locate information efficiently and with a greater degree of flexibility (Cayzer 2004, p. 52). The desegregating of semantics from content enables the team to more easily monitor emerging trends and issues.
Measure of influence in the prototype:	The frequency by which participants use the tagging mechanism and the semantic structure it generates are two indicators of dynamic semantics' influence within the prototype. Usage patterns were measured by tracking the overall and specific growth rates of tags during the test. Once the testing was complete the semantic structure formed by these tags was visualised and reviewed to determine whether it conveyed a relevant picture of events during the testing period.
Inferred influence:	A prototype embodying the principle of dynamic semantics will allow participants to generate a semantic structure that reflects the specific events and needs of the project. Assuming an efficient interface this semantic model will grow quickly and will act as an accurate indicator of key collaboration events.

7.2.7. Testing the principle of decentralisation

Principle:	Decentralisation
Principle's intent:	A collaboration system should not rely on a single entity for its existence. Instead data should be distributed across and accessible from the broadest practical number of locations.
General effect on technical design:	The technical implementation should promote the use of multiple, independent systems in partnership to ensure collaboration redundancy (Erenkrantz & Taylor 2003, p.3). Within this environment transferring data to alternate systems should occur without impacting performance or cohesiveness.
Prototype functionality:	The prototype was Web-based so that participants could interact with it from any Internet-connected location or device. The prototype allowed participants to use two systems to describe design; a loosely structured blog and a file-centric 'web area'. Each participant was free to decide how information was distributed across these two components.
General effect on collaboration:	Removing access barriers encourages interaction and provides more freedom in the way users can express ideas or utilise information (Berridge & Brown 2002, p.488-490). Promoting data portability and distribution improves the resilience of a project's knowledge to changes in the team's composition.
Measure of influence in the prototype:	If participants used the prototype's blogging and 'web area' components differently, but on a regular basis, decentralisation promoted frequent, flexible collaboration. The composition of content submitted to the blogging and 'web area' components was compared to understand if the quantity and types of data submitted to each differed. The time content was published was used to identify when the prototype was in use. Usage habits of users was assessed by questionnaire on completion of the test.
Inferred influence:	A prototype embodying the principle of decentralisation will be more frequently interacted with because access is not restricted to a specific location. Internally decoupling the prototype's functional components will provide participants with more choice and flexibility in the way they communicate ideas.

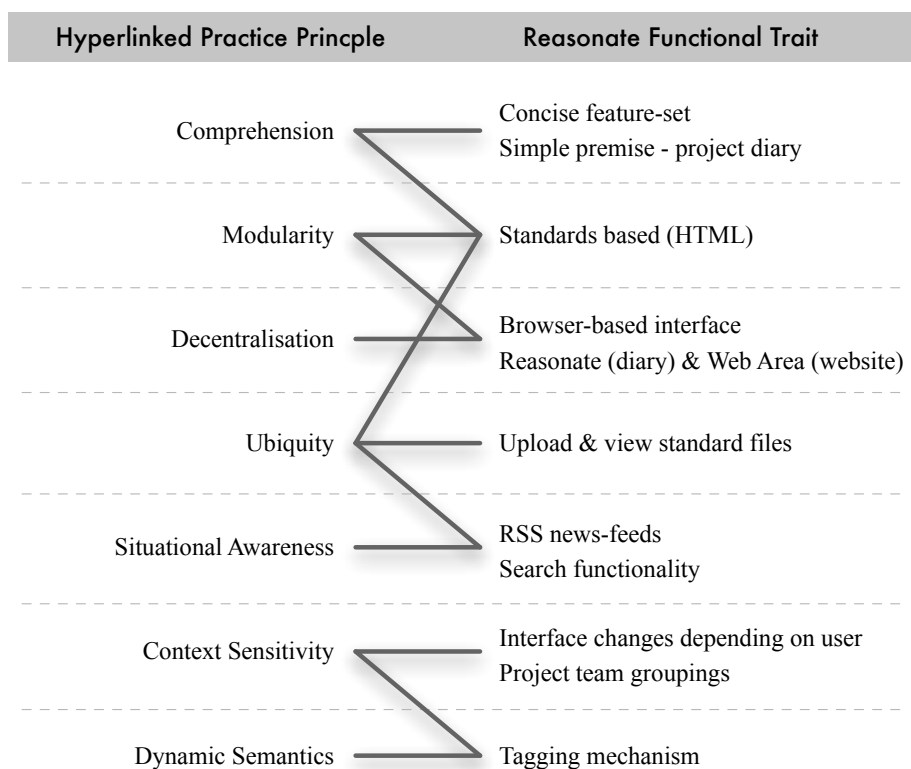
7.3. The design of the Reasonate prototype

Reasonate’s underlying purpose was to determine whether a digital tool that embodied the principles of Hyperlinked Practice would promote the recording of Building Stories. With this objective in mind, the the BBS303 ‘Digital Craft’ course was an ideal testing environment, because students were encouraged to keep a record of their digital collaboration processes. To assist students in meeting this goal, Reasonate’s core functionality centred around that of a weblog (blog). The blog metaphor was chosen because at the time it was the simplest and most ubiquitous means of publishing personal thoughts on the Internet (Nardi, Schiano & Gumbrecht, 2004, p. 222). At the end of the testing process a survey was conducted to gather the students’ opinions on the use of Reasonate and its collaboration influence. Many of the questions were designed to measure the hypothesised collaboration effects of the Hyperlinked Practice principles.

7.3.1. The underlying design motives

The composition of Reasonate’s functionality was driven by the seven principles of Hyperlinked Practice and the practical requirements of the University course. The mapping of the principles to their derived functionality is illustrated in Figure 7.3.

Figure 7.3: Reasonate functionality derived from the principles of Hyperlinked Practice



All of the principles of Hyperlinked Practice are evidenced, to some degree, within a blog:

- **Situational Awareness** - Blogs usually generate RSS feeds of new content.
- **Ubiquity** - Blogs are now a mainstream communication medium.
- **Comprehension** - As the digital equivalent of a diary, it is easy to understand.
- **Context Sensitivity** - Like a diary, blog content is related to a specific date.
- **Emulative Modularity** - The functionality of a blog can be easily replicated.
- **Emotive Semantics** - Most blogs support ‘tagging’ to categorise content.
- **Decentralisation** - It is stored in one location, but can be read by any web browser.

Although the principles are evident to some degree, collaboratively recording the Building Story imposes unique demands that are not met by a conventional blogging system:

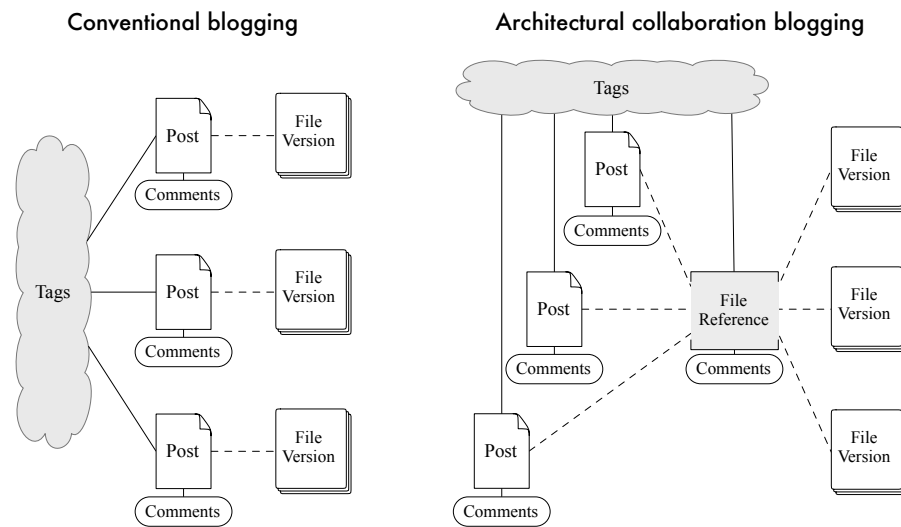
7.3.1.i. Team-centric blogging (context sensitivity)

Conventional blogs can have multiple contributors, but generally users cannot be organised into project teams that collaboratively publish content related to a specific field of interest. An digital collaboration blog needs this capability, because it provides the contextual information necessary to inform the Building Story, and enables the user interface to be tailored to meet the specific needs of the project team.

7.3.1.ii. Equality between blog posts and digital files (decentralisation)

The digital files uploaded to Reasonate are the project’s artefacts, and its blog entries describe the activities that occurred. When combined, these two equally important elements comprise the events of a Building Story. However, within a conventional blog this relationship is not equal, because digital files rely on being linked to by posts before they can become part of the Building Story. To address this imbalance, a digital collaboration blog should instead treat digital files as independent entities that exist as part of the Building Story. In support of this concept, each unique digital file was given a dedicated page within Reasonate where the file could be tagged and commented upon independently by any blog posts that reference them. This functional trait, illustrated in Figure 7.4, worked to ensure that Reasonate’s digital files were treated as equal and independent entities to the blog posts which referenced them.

Figure 7.4: The relationship between blog posts and files within a conventional blog and one optimised for digital collaboration

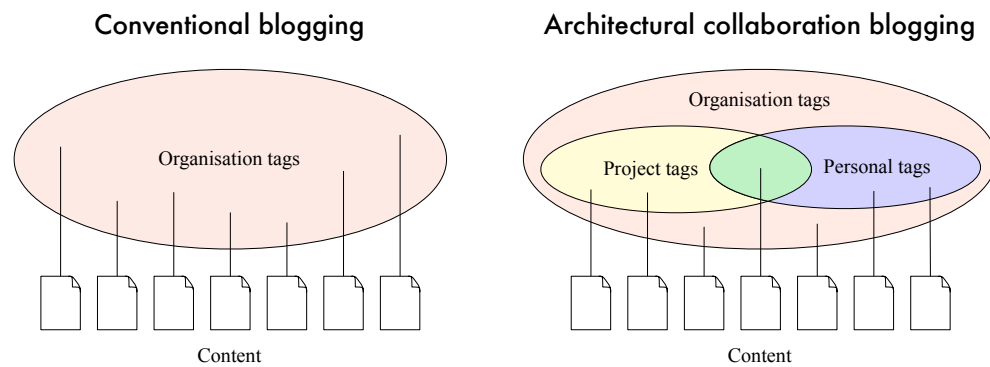


A project’s history is often reflected within the different historical versions of a single digital file. For example, as the digital model evolves the design conversation associated with it shifts from conceptual issues through to specific architectural details. During this process the digital model is the link between these two conversational threads, and it is therefore important that this point of reference is preserved. To achieve this, digital files uploaded to Reasonate were compared with others already on the system in order to determine uniqueness. If the file was an updated version of one already present within the system, a relationship between the two different versions was established. A conceptual overview of this versioning system, and the relationships between the versions of a file and the blog posts that reference them, is illustrated in Figure 7.4.

7.3.1.iii. Multi-tiered tagging (emotive semantics)

A conventional blogging platform usually employs a tagging mechanism for categorising content. As illustrated in Figure 7.5, these systems do not generally distinguish between tags assigned by the user, their project team, or the broader organisation. This poses a problem when recording the Building Story, because a tag can have a different meaning depending on who applies it, or its intended audience. Comprehending and leveraging these nuances is important when working on projects, because often the semantic structures employed within different teams will vary considerably. To address this concern, Reasonate employed a multi-tiered tagging mechanism that differentiated between personal, project and organisational tags. This mechanism enabled content to be navigated more efficiently, because students better understood what team members had tagged.

Figure 7.5: The sets of tags present within a conventional blog and one optimised for digital collaboration



7.3.1.iv. Overall cohesiveness (comprehension)

In theory, Reasonate's core functionality could be provided by a number of existing software tools and services that were tightly integrated. This would have resulted in a more modular and decentralised collaboration system, but given the limited time and resources available for development, this was approach impractical. Integrating multiple, different systems is a complex process, and if not undertaken correctly the resulting user experience can be inconsistent. Rather than taking this risk, Reasonate was developed as a single system which had consistently designed and clearly presented functionality that was easy to comprehend.

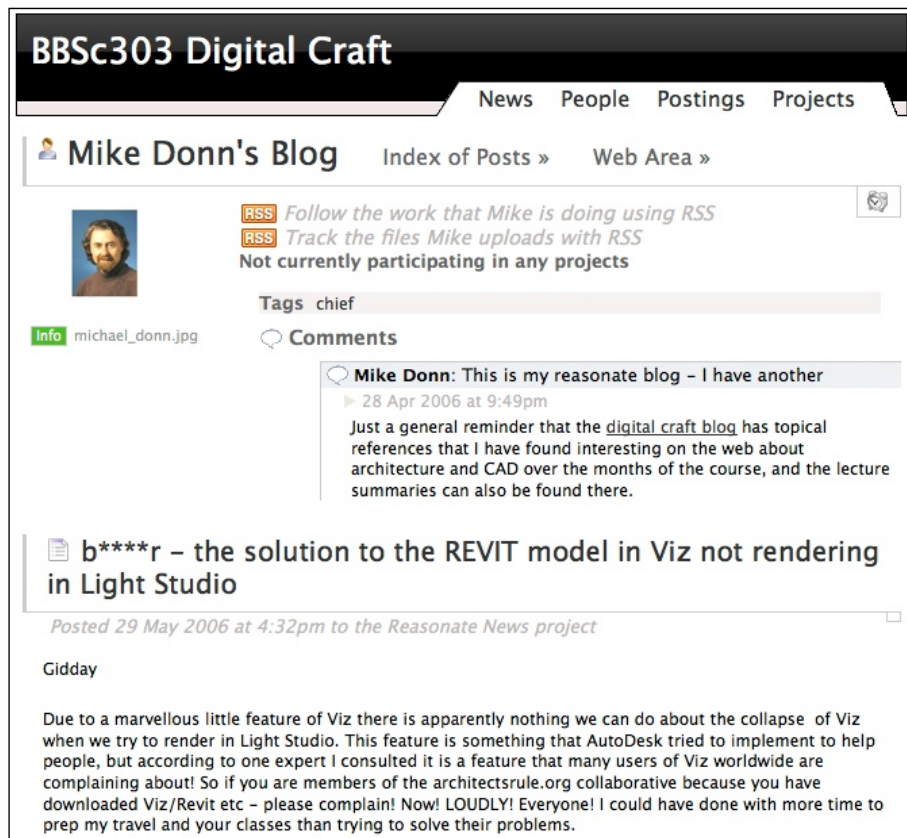
7.3.2. Core functionality

The underlying design concepts within Reasonate were reflected by eight pieces of core functionality. These functions covered the creation, organising, searching and broadcasting of content by project teams. In addition, a questionnaire was integrated to capture student opinions on Reasonate usage patterns, and its influence on collaboration.

7.3.2.i. User identities and personal blogs

All students, tutors and course coordinators were assigned Reasonate user accounts. On logging in, users were presented with their personal blog, an example of which is illustrated in Figure 7.6. This blogging area was used to present content that was not associated with project work. Examples of this included tutorial exercises, technical questions, and news related to the overall course.

Figure 7.6: A personal blog within Reasonate



7.3.2.ii. Publishing blog posts and commenting

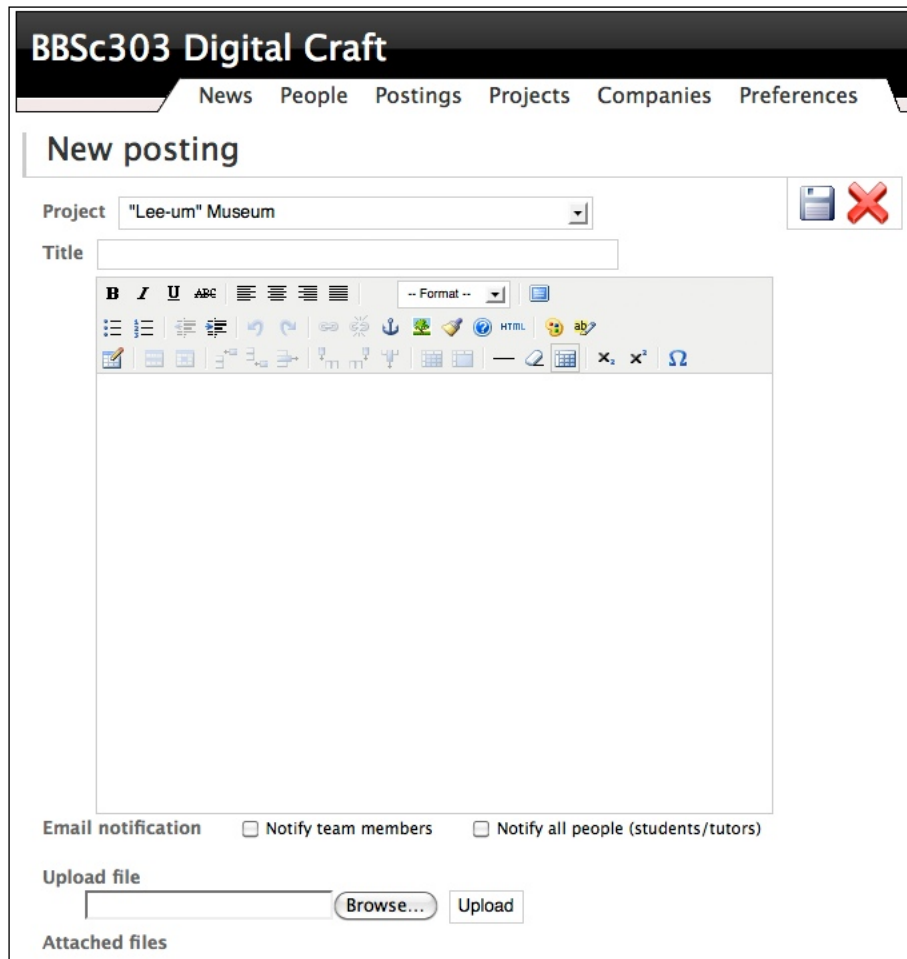
Reasonate's primary piece of functionality was the ability for users to publish blog posts. The interface for this task, illustrated in Figure 7.7, was simple and mirrored that of conventional blogging tools. Using a drop-down field at the top of the screen, users could select whether to publish the post to their project or personal blog spaces. Along with the blog text, users could upload and associate multiple digital files with the post. Finally, users could have team members and supervisors automatically notified that the blog post had been published via email.

Students were unable to delete or modify blog posts once published. These options were restricted because Reasonate was a permanent diary. Deleting entries would destroy part of this record, and the ability to modify posts would have encouraged students to rewrite history. Mistakes did sometimes occur during testing which led to some posts being published prematurely and without important content. In these rare cases, students could contact a supervisor to have the erroneous post modified or removed.

Blog posts, digital files and project pages could be commented on by any Reasonate user. To encourage two-way conversations, the interface for publishing a comment was

streamlined. When a new comment was published, the author of the original content and any contributors to the conversation thread were automatically sent an email notification.

Figure 7.7: The blog publishing interface within Reasonate



7.3.2.iii. File attachments and versioning

Selecting the 'Info' link below any digital file uploaded to Reasonate displayed further information about the resource. Each unique digital file was assigned its own webpage within Reasonate where users could assign tags, write comments and view the blog posts which referenced it. An example of this screen, showing a digital file with two different versions, is illustrated in Figure 7.8.

The uniqueness of a digital file was determined by a combination of its filename, the user who uploaded it, and the project it was assigned to. A file that was not associated with a project was unique if the user had not previously uploaded a file of the same name. If the file was associated with a project, Reasonate searched through all the files associated with the team to establish whether any of them had the same name. If this search

discovered that a file of this name already existed within Reasonate, then the uploaded file was considered an updated version. If this was the case, Reasonate automatically formed a relationship between these two files, to allow their history to be better understood.

Figure 7.8: Reasonate's file information screen showing a file's different versions



7.3.2.iv. Project teams and content aggregation

Reasonate users were able to easily create and join project groups. Once a member of a project, users could select to publish content to it rather than their personal blog. Reasonate aggregated these contributions and presented them on the project's blog. Tags added to project content by team members were presented as a separate list, alongside the user's personal tags and those applied by every other Reasonate user. This differentiation allowed team members to develop a project-specific semantic structure, that they could leverage to organise and navigate their project team's work.

7.3.2.v. Rich searching of content

Reasonate had a basic search field located at the top right of every page that allowed users to quickly perform text-based content searches. In addition to this, an advanced search form, illustrated in Figure 7.9, was available for more complex searches. Using this advanced search form, a number of filters could be applied to searches:

- Content contributed by a specific user or project team.
- Digital files of a certain type, for example 3D Models.
- Content published within a defined period of time.

Figure 7.9: Reasonate's advanced search form

The screenshot shows the 'Advanced Search' form for 'BBS303 Digital Craft'. The page has a dark header with the site name and navigation links for 'News', 'People', 'Postings', and 'Projects'. The search form includes a 'Search term or tag' input field with a 'Search' button. Below this are three filter sections: 'Look in a certain place' with radio buttons for 'Entire site', 'This person's work:' (with an input field), and 'Project:' (with a dropdown menu showing '"Lee-um" Museum'); 'Look for specific media' with radio buttons for 'Any media' and 'The specific media identified below:', followed by checkboxes for '3D Model', 'Image', 'Sound/Music', 'Archive', 'PDF', 'Spreadsheet', 'Document', 'Presentation', and 'Video'; and 'Limit search to a certain time' with radio buttons for 'Any time', 'After this date:' (with a date input and calendar icon), 'Before this date:' (with a date input and calendar icon), and 'Between:' (with two date inputs and calendar icons). A final 'Search' button is located at the bottom right of the form.

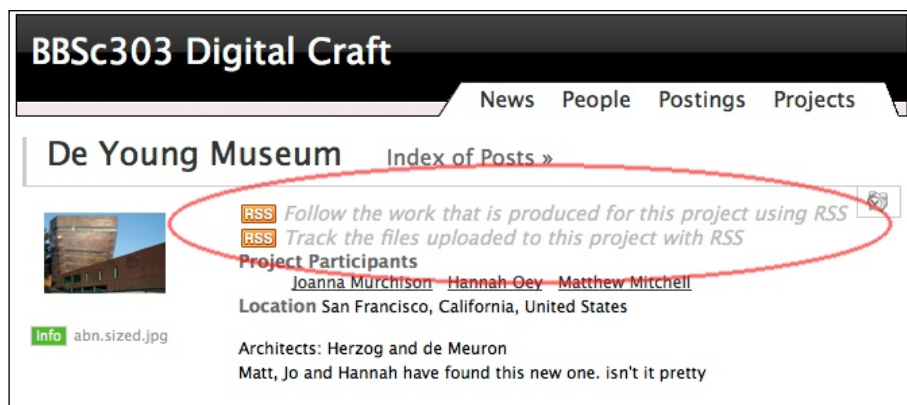
7.3.2.vi. News-feeds of latest content

Really Simple Syndication (RSS) provided a means of automatically monitoring new Reasonate content by third-party applications, the most common being news-readers. Reasonate generated RSS feeds for a variety of subject areas:

- New posts published by any user.
- New posts published by a specific user.
- New digital files uploaded by a specific user.
- New posts published to a project.
- New digital files uploaded to a project team.
- New search results for a search query, for example ‘3D Models’.

Throughout the Reasonate interface the orange RSS logo and supporting text, as illustrated in Figure 7.10, indicated the availability of a news-feed of content changes. Unfortunately, unforeseen limitations on what software could be installed on the Victoria University computers meant that the majority of students were unable to leverage these RSS feeds to monitor Reasonate activity.

Figure 7.10: Examples of the RSS icons and text within Reasonate’s interface

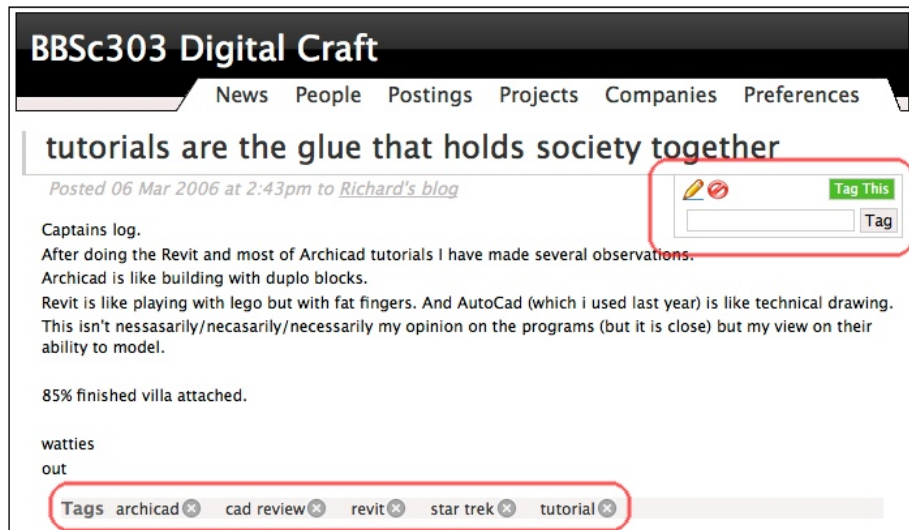


7.3.2.vii. Content tagging at the person, team and organisation level

Any blog post, digital file, personal blog or project team could be tagged multiple times by Reasonate users. Tagging allowed users to organise content into meaningful structures for later reference. To encourage tagging, the process was made as simple and streamlined as possible. Pressing the ‘Tag This’ button located below the content’s title, illustrated in Figure 7.11, presented the user with a small text-field. Here they could type the name of the tag and assign it to the content. The tags associated with a piece of content were listed

in the content footer, as illustrated in Figure 7.11. Users could delete their own tags by pressing the 'X' icon to the right of the respective term.

Figure 7.11: The 'add tag' and 'tag listing' interfaces for a Reasonate blog post



Lists of the tags employed by the user, their project team, and the entire class were available on the right-hand side of all Reasonate screens. Users could select to view these lists as simple lists, or as 'tag clouds' where more frequently used terms were presented in a larger font (Sinclair & Cardew-Hall, 2008, p. 17). Selecting a tag from one of these lists would present the user content tagged with the term, as illustrated in Figure 7.12. To the right of these results were related tags, which were identified by analysing what tags had been used in conjunction with the selected term.

Figure 7.12: Reasonate's tag search screen and personal "tag cloud"



7.3.2.viii. The Web Area and personal websites

As part of the BBS303 'Digital Craft' course requirements, students were required to publish their own personal websites that showcased their team's digital model and renderings. This presented an opportunity to decentralise the Building Stories captured within Reasonate by providing two different communication experiences:

- A consistent and tightly controlled blogging system for recording the process.
- A personal area where students could publish highly customised and refined websites.

Both of these sections helped to shape the team's Building Story. The blogs were a chronological account of actions and decisions, whilst the personal website presented a summary of the overall process and its outcomes. These two different experiences could be woven together using hyperlinks embedded within the content of both sections.

To meet these objectives, Reasonate had a Web Area where students could easily upload an entire personal website. The mechanism used for this upload process was WebDAV (Whitehead & Goland, 1999), because it allowed the Web Area to be integrated into Dreamweaver, the website authoring tool used within the course. Once configured, students could publish their website to Reasonate's Web Area with a single command. Within Reasonate, each personal blog had a link to the respective user's Web Area. If the Web Area contained a file named 'index.html', selecting this link would automatically display the student's personal website. If this file did not exist, or was named incorrectly, Reasonate displayed a help screen and simple file browser to assist the student in configuring WebDAV and troubleshooting website technical problems.

7.3.2.ix. Feedback questionnaire

The Reasonate user survey was conducted using a questionnaire system integrated into Reasonate, as illustrated in Figure 7.13. The questionnaire functionality was only made available in the last few weeks of testing, as it was intended to record the student's overall experience and opinion of Reasonate and its influence on collaboration.

Figure 7.13: Reasonate’s questionnaire interface for conducting the user survey

The screenshot shows a web interface for a questionnaire titled "Reasonate Questionnaire" under the heading "BBS303 Digital Craft". At the top, there is a navigation menu with links for "News", "People", "Postings", "Projects", "Companies", and "Preferences". Below the title, a paragraph explains that the questionnaire is anonymous and that users should not resubmit their answers. The section "Past Experience" contains three questions:

- 1. Prior to the BBS303 course how would you rate your level of experience with using the Internet?**
This question has a horizontal scale with five radio buttons labeled 1 through 5. Below the scale, "No experience" is aligned with 1, "Average" with 3, and "Excellent" with 5.
- 2. Prior to the BBS303 course did you regularly read any weblogs?**
This question has two radio buttons labeled "Yes" and "No".
- 3. Have you ever created your own web-page or blog entry prior to this course?**
This question has two radio buttons labeled "Yes" and "No".

7.3.3. The Reasonate use and collaboration influence questionnaire

During the final stages of the testing period, the students were asked to complete a questionnaire to establish their prior technical experience, their use of Reasonate, and its perceived influence on the collaboration process. These questions and their underlying research intentions are described in Table 7.2. Comprehension and situational awareness were the focus of many questions, because the effects of these principles are linked to the participant’s ability to understand and utilise the digital tools, collaboration processes, and project information.

The pretested questionnaire was made available to students as a digital form within Reasonate. To ease the submission and analysis process, all of the questions had single or multiple choice responses. An analysis of the relative frequency with which the different response options were chosen was conducted for each question. These results informed the evaluation of Reasonate, because they provided insight into the student’s prior experience, usage patterns, and their perception of the system’s usefulness during collaboration. In conjunction with the analytical data collected during testing, this understanding helped to identify what effect, if any, the principles of Hyperlinked Practice had on the operation of the software prototype and collaboration.

Table 7.2: The intention of the survey questions and their contribution to the analysis of the principles of Hyperlinked Practice

1. Prior to the BBS303 course, how would you rate your level of experience with using the Internet?	
Possible responses:	Scale between 1 to 5, where 1 is 'no experience' and 5 is 'excellent'
Principle tested:	Comprehension
Question intention:	<i>(Questions 1, 2 and 3)</i> To provide insight to the student's experience and use of Internet-based media prior to the course. This information was compared to the usage data recorded during the initial stages of Reasonate testing to build a more complete understanding of adoption patterns. The principle of comprehension was demonstrated if Reasonate adoption and usage rates were high, irrespective of the experience of those using the tool.
2. Prior to the BBS303 course did you regularly read any weblogs?	
Possible responses:	Yes or no
Principle tested:	Comprehension
Question intention:	<i>Refer to the intention of question 1.</i>
3. Have you ever created your own webpage or blog entry prior to this course?	
Possible responses:	Yes or no
Principle tested:	Comprehension
Question intention:	<i>Refer to the intention of question 1.</i>
4. How frequently did you use Reasonate to review your project or other people's work?	
Possible responses:	Scale between 1 and 5, where 1 is 'never' and 5 is 'as often as possible'
Principle tested:	Situational awareness
Question intention:	To determine how often students reviewed Reasonate content during the project work. This indicated whether situational awareness was promoted through personal reflection.

5. In what locations did you access Reasonate either to view or post new work?	
Possible responses:	Multiple choice: <ul style="list-style-type: none"> • Within the Architecture School computer labs • In other computing facilities provided by Victoria University • At your place of residence • Within a professional workplace • Other location such as a cafe, web kiosk or friend/relative's house
Principles tested:	Decentralisation and ubiquity
Question intention:	To identify which general environments students accessed Reasonate from. The principle of decentralisation was demonstrated if the most participants accessed Reasonate from many different locations. In addition this question assessed the influence of ubiquity, because this level of flexibility can only be achieved when the underlying technologies used to interact with the service are ubiquitous.
6. In your own opinion how frequently did you post new blog entries on Reasonate?	
Possible responses:	Scale between 1 and 5, where 1 is 'never' and 5 is 'as often as possible'
Principle tested:	Comprehension
Question intention:	The student's opinion on the frequency of their contributions to Reasonate can be validated by the recorded analytical usage patterns. The perception of frequency posting blogs would suggest a high adoption rate. High adoption rates would demonstrate the principle of comprehension.
7. When posting content on average how long was spent writing and proofing the post?	
Possible responses:	Multiple choice: <ul style="list-style-type: none"> • 1-5 minutes • 5-10 minutes • 10-20 minutes • 20-30 minutes • More than 30 minutes
Principle tested:	Comprehension
Question intention:	The time invested in composing content indicated what influence contributing to Reasonate had on the student's workflow. Although not a specific demonstration of the principle of comprehension, the time required for contributions can influence their frequency.

8. Did the act of documenting your progress on Reasonate encourage further critical thinking on the work you had undertaken?	
Possible responses:	Scale between 1 and 5, where 1 is 'not at all' and 5 is 'always'
Principle tested:	Situational awareness
Question intention:	<i>(Questions 8 and 9)</i> Reflection is an important factor in the development of situational awareness, because it reinforces an understanding of the events and decisions which occurred. The ability of Reasonate to stimulate this activity is a demonstration of the principle of situational awareness.
9. Did the act of reading and writing these blogs provide extra motivation in the course?	
Possible responses:	Scale between -2 and 2, where -2 is 'strong negative effect' and 2 is 'strong positive effect'
Principle tested:	Situational awareness
Question intention:	<i>Refer to the intention of question 8.</i>
10. How frequently did you use the tagging functionality of Reasonate?	
Possible responses:	Scale between 1 and 5, where 1 is 'not at all' and 5 is 'always'
Principle tested:	Emotive semantics
Question intention:	The student's opinion on how frequently they tagged Reasonate content supported the recorded analytical usage patterns. High adoption rates would demonstrate the principle of emotive semantics.
11. For what reasons did you tag content you or someone else had posted to Reasonate?	
Possible responses:	Multiple choice: <ul style="list-style-type: none"> • After lectures about tagging • When prompted by team members • When prompted by tutors/course co-ordinators • To find the post at a later date • To help organise the work I was submitting
Principles tested:	Emotive semantics
Question intention:	Measuring the influence of the principle of emotive semantics requires an understanding of why content was tagged. Tagging will be used voluntarily and in partnership with other team members within a system that embodies the principle of emotive semantics.

12. How often did you use the RSS functionality provided in Reasonate?	
Possible responses:	Single choice, based on a series of statements: <ul style="list-style-type: none"> • I did not try using this functionality • I tested the service but did not see reason for using it further • I tested the service but technical reasons stopped me from using it • I used the functionality occasionally • I used the functionality frequently
Principle tested:	Situational awareness
Question intention:	To determine whether the students tried to use Reasonate's RSS functionality, and if so, how often they used it to monitor the contributions of team members and others within the class.
13. How often did you use the search functionality provided in Reasonate?	
Possible responses:	Single choice, based on a series of statements: <ul style="list-style-type: none"> • I did not try using this functionality • I tested the functionality but did not see reason for using it further • I used the functionality occasionally • I used the functionality frequently • I would have liked to search more but the results were no relevant
Principle tested:	Situational awareness
Question intention:	To determine whether the student's leveraged Reasonate's search capability in order locate relevant content. Search enables the location of content, and by extension promotes situational awareness.
14. How did Reasonate effect your experience of the BBS303 course as a whole?	
Possible responses:	Scale between -2 and 2, where -2 is 'strong negative effect' and 2 is 'strong positive effect'
Question intention:	<i>(Question 14 and 15)</i> To measure Reasonate's influence on the student's experience of the course and their ability to collaborate. This information was not directly related to the testing of any specific principle, but it did provide an indication of the general effect of a collaboration tool embodying the principles of Hyperlinked Practice.
15. In your opinion, what effect did Reasonate have on your team's ability to work effectively?	
Possible responses:	Scale between -2 and 2, where -2 is 'strong negative effect' and 2 is 'strong positive effect'
Question intention:	<i>Refer to the intention of question 14.</i>

7.4. Constructing the Reasonate prototype

The Reasonate prototype was purpose-built because no existing commercial or open-source software package encompassed all of the required functionality. An agile approach was followed during development, because it allowed the user's needs and feedback to be rapidly accommodated. As a result of this development strategy, many changes were being made to the prototype during the testing period. A year after Reasonate testing, its core functional concepts were independently reimplemented within the same university course. Although the resulting system was not functionally identical, the process illustrated the ability of the digital collaboration tool to be emulated, and provided usage statistics that helped to identify the replicability of Reasonate's collaboration effects.

7.4.1. A purpose built prototype

Reasonate was purpose built to satisfy the testing requirements of the Hyperlinked Practice principles, and the functional needs of the BBS303 'Digital Craft' course. At the time of testing, no commercial or open source software package was available that possessed all of the functionality required. Viable software candidates were available that satisfied some functional aspects, but meeting all of the prototype's needs would have required the integration and support of two, or more, third-party software packages. The risks and limitations of this approach were such that a purpose-built prototype was more appropriate.

7.4.2. The influence of agile development

An agile software development process (Abrahamsson, Salo, Ronkainen & Warsta, 2002) was applied throughout the development and testing of Reasonate. During this time, many prototype iterations were produced, and feedback from students and supervisors was frequently incorporated. This development strategy was appropriate because the prototype had "an accelerated time schedule combined with significant risk and uncertainty that generated constant change during the project" (Highsmith, 2002, p. xxii). Ultimately the purpose of the prototype was to test whether a digital tool that embodied the principles of Hyperlinked Practice promoted the recording of Building Stories. Finalising the prototype's functionality prior to the four month testing process would have stunted development, and potentially impaired its ability to record Building Stories. In contrast, modifying the prototype during testing enabled the lessons learnt in practice to be applied

immediately. This ensured that the performance of the prototype, the effect of the principles, and their combined ability to record the Building Story was thoroughly tested.

The deployment schedule of new functionality during testing was primarily driven by course requirements. This gradual rollout of functionality eased the learning curve and supported testing of the principle of comprehension, which states that users will make better use of a tool whose purpose and functionality is readily understood. From a development standpoint this phased deployment provided more time to refine functionality in response to usage patterns. This ensured that the functionality being tested was responding to the student's actual needs, rather than assumptions made prior to the course.

During testing, users were encouraged to provide feedback via a number of means:

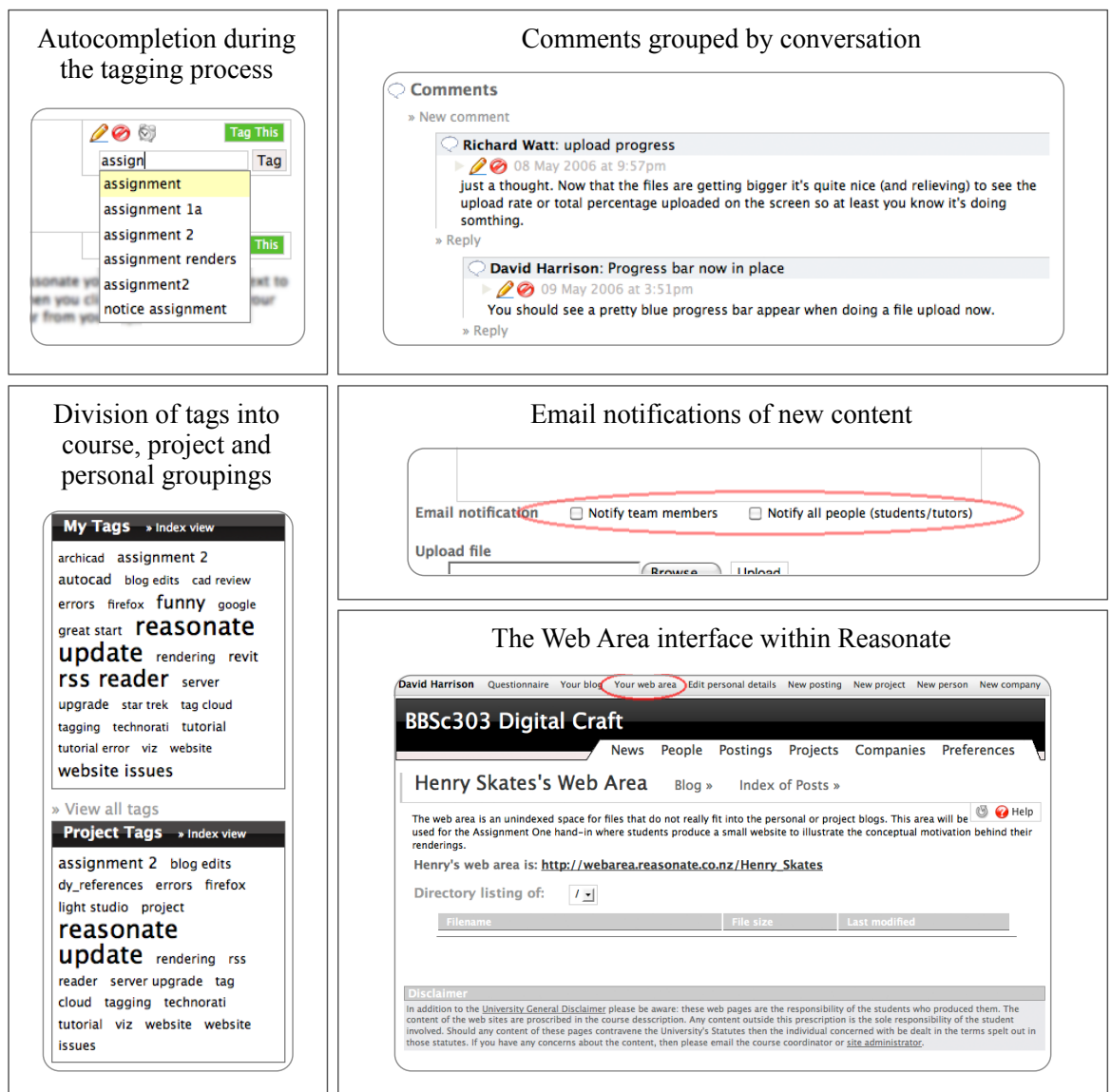
- **Lectures** - Reasonate functionality was introduced to students via a series of small lecture presentations. At the end of these presentations students were invited to provide feedback. Additionally, mid-way through testing a feedback session was conducted so that students could provide suggestions about the features of Reasonate and raise questions about its use.
- **Tutorial sessions** - Students could discuss Reasonate problems with tutors and coordinators during tutorial sessions held twice a week.
- **Reasonate** - Students were encouraged to publish feature requests and feedback to Reasonate, where they could be openly discussed and monitored.

All feedback received was reviewed and responded to in a timely manner. In some cases the requests were practically or technically too demanding to justify implementing.

7.4.3. The influence of user feedback

During Reasonate testing, feedback from users resulted in numerous functionality refinements and additions. Examples of the most significant improvements in functionality based on user feedback are illustrated in Figure 7.14. The majority of these improvements focused on the clearer presentation of information. The addition of email notifications and improvements to the Web Area interface were also made to ensure the students could reliably communicate with each other.

Figure 7.14: An overview of Reasonate functionality driven by user feedback



7.4.3.i. Autocompletion during the tagging process

The initial interface for tagging Reasonate content was a simple text field. This posed a number of difficulties for users who were trying to construct a cohesive semantic structure:

- **Recalling** - Without a visual reference, users struggled to remember the tags which had previously been applied to similar content.
- **Consistency** - The grouping of content by tag-based semantic structures depends on the consistent spelling of terms. Without an autocomplete feature, the number of spelling derivatives quickly multiplied. This diluted the semantic structure and made browsing it less efficient.
- **Efficiency** - Repeatedly typing in the tag was time consuming and viewed as a potential barrier to use.

To address these issues, the tagging interface was modified to present tag suggestions below the text field. Users could select one of these tags, or continue typing the name of a new tag. Tag suggestions were based on a simple, text-based comparison of the tag field. A more intelligent, semantic-based suggestion system was not employed due to time and resource limitations.

7.4.3.ii. Division of tags into course, project and personal groupings

Initially the tags assigned by users were presented on the right-hand side of the Reasonate interface as a single list or tag cloud. As the number of tags applied by students grew, this listing became too long and unwieldy to use as a quick reference. To compensate, this tag listing was divided into three different groups:

- **Course** - All of the tags assigned to any Reasonate content.
- **Project** - Tags used within the projects a user was associated with.
- **Personal** - Tags that the user had assigned to any Reasonate content.

Modifying the presentation of tag listings in this manner allowed users to focus on the semantic structures they were using personally, or what was being used by their team. Providing more granular views limited the proliferation of tag derivatives because users were prominently reminded of the tags they and their team had already applied.

7.4.3.iii. Comments grouped into conversation threads

Comments made on Reasonate content were initially presented in chronological order. For the majority of situations, where only a few comments were made, this ordering mechanism was acceptable. However, in cases where comment-based discussion became intense, this simple ordering mechanism made comprehending the different conversation threads difficult. To resolve this shortcoming, the comments mechanism was modified so that comments could be presented as conversation threads. Changing Reasonate in this manner improved the recording of the Building Story, because it linked conversations on the same subject together, which allowed students to comprehend and reference relevant content in a timely fashion.

7.4.3.iv. Email notifications for new content

When initially deployed Reasonate had no functionality for directly notifying users of new content via email. At the time this did not seem necessary because users could subscribe to RSS news-feeds and easily review new content within the web interface. Unfortunately, in practice this proved inadequate for a number of reasons:

- **Software deployment** - At the time of testing, most RSS news-feed readers were third-party applications or web browser extensions. Unfortunately, third-party applications could not be installed onto the majority of the University's computers. This meant most students could not leverage Reasonate's RSS functionality.
- **The assumption of attention** - Many students assumed that published Reasonate content would be read by all relevant parties. Often this was not the case because recipients were busy, or focused on other content. The course coordinator was most afflicted by this problem, and it led to situations where students felt wronged because the problems they had documented were not appropriately acknowledged.
- **Recording vs communicating** - Students tended to use Reasonate as a recording tool rather than a communications device. It was therefore common for students to document an activity in Reasonate and then compose an email to their colleagues containing exactly the same content. Students expressed displeasure at this double-handling of progress reports, especially as many did not realise they could include a hyperlink to the Reasonate content within their email.

To address these shortcomings, an option was provided at the bottom of Reasonate's 'new post' screen to automatically email team members and supervisors. This provided a direct communications mechanism that ensured content was acknowledged without imposing a further burden on the author.

7.4.3.v. The Web Area interface within Reasonate

Students were provided with a WebDAV file upload interface that allowed them to publish the websites they had created to the Reasonate web server. Uploading this content was a complex process that met some obstacles:

- **Configuration** - Entering the correct WebDAV settings into the website authoring software (Dreamweaver) was error prone.
- **Conventions** - Many students incorrectly named important website files, for example index.html. This resulted in their websites failing to work as expected.
- **Cohesiveness** - There was no obvious link between student's work in Reasonate and the websites which they had developed.

To address these issues a Web Area tab was added to the Reasonate web interface. This provided a number of functions:

- A description of the relevant WebDAV settings to use within the website authoring software (Dreamweaver) so that the upload would work correctly.
- A visual listing of the files uploaded to the Web Area. This allowed students, tutors and staff to easily troubleshoot misbehaving websites.
- A direct link from the student's Reasonate blog to their customised website. This created a more coherent and convenient browsing experience.

These changes improved the comprehension of Reasonate, because it enabled students to utilise the functionality that was available. Furthermore, these interface modifications promoted decentralisation by linking Reasonate's blogging and Web Area components.

7.4.4. Independent reimplementations of the prototype

In 2007 the underlying functionality of Reasonate was independently reimplemented for use within the same BBS303 'Digital Craft' paper by course coordinator Michael Donn. Reasonate's basic functionality was replicated using Google Blogger, a standard blogging platform, in partnership with a Google Groups discussion forum, and an FTP server based in Victoria University. An example of the Blogger implementation for BBS303 in 2007 is illustrated in Figure 7.15. The mapping of the Reasonate functionality to that of the Blogger-based system is described in Table 7.3.

Although this system was not functionally identical to Reasonate, it did demonstrate that the collaboration system could be emulated in a modular manner. The data recorded within the two systems could be directly compared because they were both applied within the same university course and under similar conditions.

Figure 7.15: The Blogger-based reimplementations of Reasonate

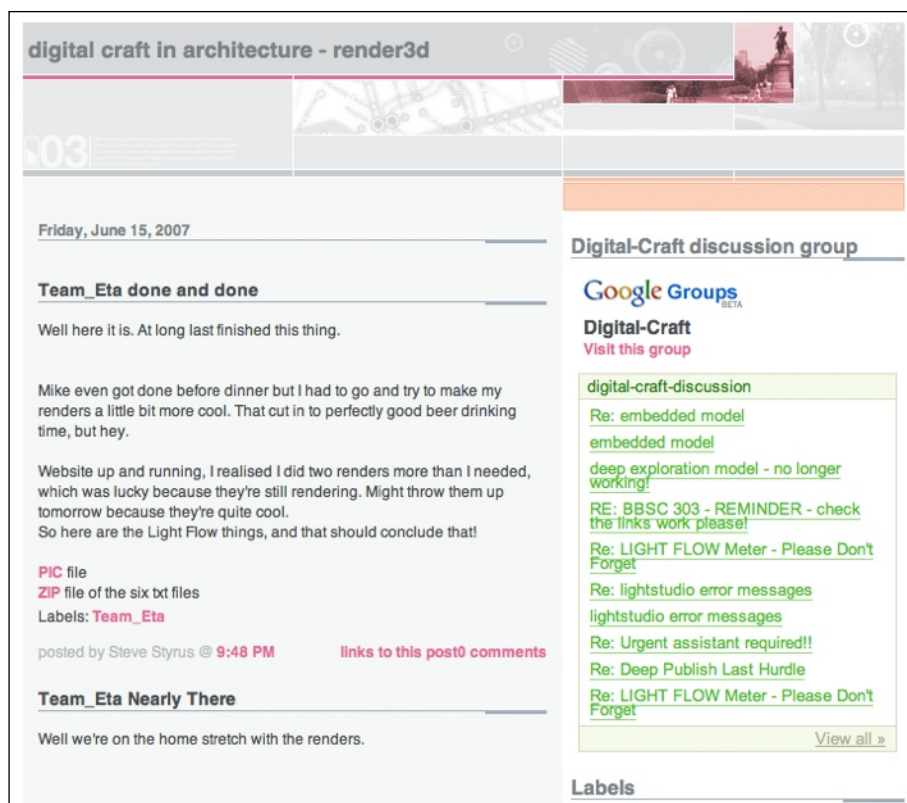


Table 7.3: A functional comparison between Reasonate (2006) and Blogger (2007)

Reasonate (2006)	Google Blogger (2007)
User identity and personal blogs.	Each student had a Google Blogger account which was used for identity purposes. Personal blogs were associated with these accounts, but because they were independent to the BBS303 2007 blog they were generally unused.
Publishing blog posts and commenting.	Blogger was used by students to publish blog posts on their team's progress. These blogs could be commented upon by other Blogger users. In addition, a BBS303 Google Group was setup to host more general discussion about the course.
File attachments and versions.	Blogger supported the simple uploading and linking to of digital files, but did not support the versioning of files. Students faced difficulties uploading large files to Blogger given its servers were not hosted within New Zealand.
Project teams and content aggregation.	A BBS303 blog was created within Blogger and all students were setup as contributors to it. Beyond the use of tags to differentiate posts, there was no digital delineation of project teams.
Rich searching of content.	Blogger content could be searched using Google's public search engine tools, but these indexes did not usually reflect content changes immediately.
News-feeds of latest content.	Blogger generated an RSS feed of the latest posts published on the BBS303 blog.
Content tagging at the person, team and organisation level.	Blogger posts could be tagged by the author of the post, but not by other students. Additionally, there was no delineation between personal, project and organisation level tags.
The Web Area which allowed student websites to be uploaded via WebDAV.	An FTP server hosted within Victoria University was used to upload the student's websites. This did not integrate with Blogger in any way, but students could create hyperlinks between the content on Blogger and personal website.

7.5. Testing Reasonate

The Reasonate prototype was tested within BBS303 'Digital Craft' course at Victoria University's School of Architecture between March and late-June of 2006. Users could interact with the prototype from any Internet-connected computer. To monitor usage patterns, nightly database snapshots were generated. Reasonate's functionality was deployed incrementally, and instruction on its use was provided by presentations, supervised tutorials and various forms of online communication.

7.5.1. Prototype deployment

For most of March 2006, Reasonate was hosted on a web server located outside of Victoria University. This allowed for more frequent software updates, but it severely impaired the experience of users who wished to upload large files to the service. Not only were file uploads slow, but Victoria University's Internet usage fees financially penalised students.

To resolve these bandwidth problems, on the 26th March the prototype was moved onto a server within the School of Architecture. This move, provided students with high-speed and free access to the service onto the University network. By this stage of testing this change had become critical, because most students were regularly uploading large digital models and high-resolution imagery. On the 25th April the server hardware running Reasonate was upgraded to address performance issues students were experiencing when performing complex operations.

7.5.2. Prototype access

Reasonate was a web-based service that could be accessed at <http://reasonate.co.nz> from any Internet-connected device. Students typically interacted with the service from web browsers running on the School of Architecture's desktop computers. Students were also encouraged to use the service from other computers and locations if the opportunity arose. No restrictions were placed on what times during the day Reasonate could be used.

7.5.3. Nightly database snapshots

During the testing period a database snapshot was automatically generated every day, just prior to midnight. These snapshots allowed the usage patterns of Reasonate to be reviewed in great detail.

7.5.4. Instructions on Reasonate use

Students, tutors and course coordinators were introduced to Reasonate's functionality through a series of presentations held at regular intervals during the semester. Support was provided during tutorial sessions, via email, and through Reasonate directly.

7.5.4.i. The introduction of core concepts and functionality

Three presentations were given during course lectures. These presentations introduced Reasonate's primary functionality:

- **28th February** - Publishing blog posts and attaching files.
- **4th April** - Tagging and RSS news-feeds.
- **30th May** - Search and the feedback questionnaire.

Presentations were accompanied by a description of the underlying collaboration motives and technologies. Often they were supported by Reasonate blog posts that reviewed the topics covered during the lecture.

7.5.4.ii. Supervised use

During the course students had two supervised tutorial sessions each week. Although these were primarily for instructing them on the use of other architectural software packages, they could also pose questions about Reasonate to tutors and supervisors.

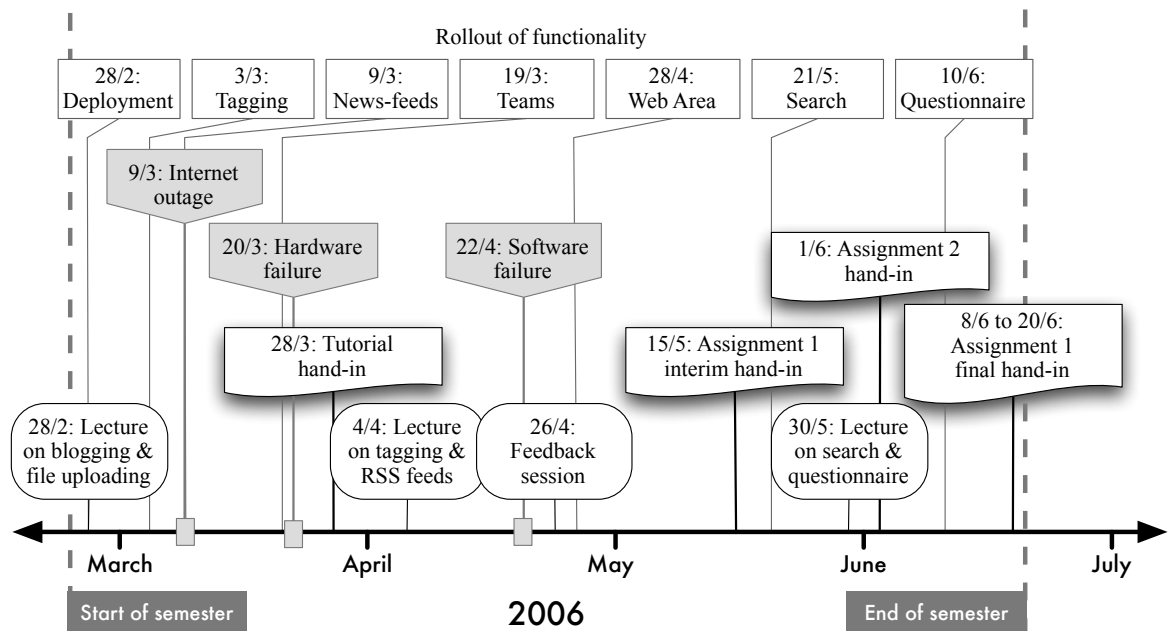
7.5.4.iii. Out of hours support

Support for Reasonate outside of tutorial hours was provided via email and through Reasonate itself. Initially the majority of questions concerning Reasonate use were received via email. However, as students became more familiar with Reasonate and its functionality, the majority of these questions were raised and answered using the service's built-in blogging and commenting tools. From the student's perspective this provided a permanent point of reference, which was important when Reasonate errors had resulted in the loss of data or the late submission of work.

7.5.5. Timeline of the testing period

Testing of Reasonate was undertaken between March and late-June of 2006. A timeline of the significant events that occurred during this time are illustrated in Figure 7.16. Overall these events were divided into four major categories: coursework submissions, deployment of functionality, instructional presentations and unplanned outages.

Figure 7.16: A timeline of the significant events during Reasonate testing



7.5.5.i. Coursework submissions

During BBS303 'Digital Craft', students had four compulsory hand-ins of coursework. Given the historical work patterns of students, it was to be expected that Reasonate activity would be more frequent leading up to these deadlines:

- **20th March** - Hand-in of the digital modelling and rendering tutorial exercises.
- **15th May** - An interim hand-in of the digital model (Assignment 1 - interim).
- **1st June** - Submission of the student's website, the digital model, and a description of the project team's activities (Assignment 2).
- **8th June** - The final submission of the light renderings from the digital model (Assignment 1 - final).

7.5.5.ii. Deployment of functionality

The deployment of core functionality occurred in stages during the testing process. This staged deployment was intended to reduce the learning curve of students, and in addition it supported the agile development process:

- **28th February** - Initial deployment of the Reasonate prototype. At this time the functionality was limited to creating blog posts, commenting and attaching files.
- **3rd March** - The ability for users to apply tags to any Reasonate content was added.
- **9th March** - RSS news-feeds for most aspects of Reasonate were made available.
- **19th March** - Project teams could be created and users could be assigned to them.
- **28th April** - The Web Area was deployed. Users could use their Reasonate credentials to upload their websites to the Web Area via WebDAV.
- **21st May** - Extensive search functionality was added so that users could better explore and understand the data contributed to Reasonate.
- **10th June** - The questionnaire functionality was made available so that students could complete a Reasonate use and collaboration experience survey.

7.5.5.iii. Instructional presentations

Four presentations were made to the students during course lectures. These presentations covered the following topics:

- **28th February** - The Reasonate prototype was introduced, and students were shown how to publish blog posts and attach digital files.
- **4th April** - An introduction to tagging content and subscribing to RSS news-feeds.
- **26th April** - An open feedback session, where students could discuss problems and suggest functional improvements to Reasonate.
- **30th May** - Students were given an overview of the search functionality, and introduced to the Reasonate use and the collaboration experience questionnaire.

7.5.5.iv. *Unplanned outages*

During the testing period there were three occasions when students could not access Reasonate due to system failures:

- **9th March** - The Internet connection for the Reasonate server went down. This resulted in students being unable to access the service for the majority of the day.
- **20th March** - The server hosting Reasonate experienced a hardware failure. The service was unavailable for most of the day whilst this problem was addressed.
- **22nd April** - The web server software crashed overnight. As a consequence, Reasonate was unavailable until midday.

In addition to these service outages, the misconfiguration of the database backup process resulted in data snapshots not being made on the following days:

- Between the 23rd March and 26th March
- 24th April
- Between the 12th May and 17th May

Although usage analytics were not recorded, Reasonate was functional during these days.

7.6. Summary

The Reasonate software prototype was designed and implemented to demonstrate the collaboration influence of the principles of Hyperlinked Practice. The principles were used to derive specific functionality within the prototype, and based on this functionality and its inferred effect, tests were identified to demonstrate each principle's collaboration influence. The software prototype was tested within a collaborative modelling class at Victoria University of Wellington. This environment was selected because it simulated relevant industry conditions, yet was insulated from many of the external forces and complexities that confound research within professional project teams. The demonstrated collaboration influence was measured using analytical data collected during the prototype's operation, and an online usage questionnaire that students completed at the end of the testing process. The next chapter details the results of this testing, and reviews these findings to determine whether the inferred collaboration influence of the principles of Hyperlinked Practice were demonstrated within the software prototype.

8. Reviewing the Performance of the Software Prototype

Analysing Reasonate use to determine the influence of the principles

To demonstrate the influence of the principles of Hyperlinked Practice, the Reasonate software prototype was tested within a collaborative digital modelling class at Victoria University of Wellington. Based on the concept of blogging, the students used the software prototype to digitally record their collaboration process and its design outcomes. Testing of the software prototype was successfully conducted between March and mid-June of 2006. During this time a large quantity of analytical usage data was collected, and a high response rate was recorded for the questionnaire undertaken at the end of the course. Combined, these results were of high enough quantity and quality to confidently review whether the principles demonstrated their intended influence within the software prototype.

Overall, the software prototype test results indicated that the principles of Hyperlinked Practice achieved their inferred collaboration effects. The influence of the principles of ubiquity, comprehension, emulative modularity and decentralisation was strongly demonstrated by the measurements that were taken during testing. The recorded influence of the principles of situational awareness and context sensitivity were not as strong. However, these test results were affected by the limited size and scope of the University course, and the relatively short timeframe of the testing process. The influence of the principle of emotive semantics was difficult to demonstrate within the Reasonate test. Primarily this was because Reasonate's tagging functionality was poorly adopted and used by the majority of students during the testing process. A year later when the Reasonate prototype was emulated by the Blogger-based service, the principle of emotive semantics had a strong influence. This suggests that the influence of emotive semantics within the software prototype was affected by limitations in the design of Reasonate's user interface, and the ability of students to perceive the immediate and long-term value of its tagging functionality.

8.1. The successful testing of the Reasonate prototype

The software prototype, Reasonate, was successfully tested within the BBS303 'Digital Craft' course at Victoria University's School of Architecture between the 1st March and 20th June 2006. Analytical usage data was collected during testing, and at the end of the course, students were asked to complete a questionnaire that investigated their use of Reasonate and its perceived collaboration influence.

8.1.1. Reasonate usage statistics

Reasonate was frequently and consistently used during the 112 day testing period. Fifty-eight students were initially part of the Reasonate testing process, but during the first few weeks six withdrew from the course. Individual Reasonate accounts were assigned to each student, the course coordinator, researcher, and the eight tutors. However, as the tutors' use of Reasonate was voluntary, their contributions were minimal.

Combined, the sixty Reasonate users published 1,818 blog posts, uploaded 2,275 digital files, and assigned 1,012 tags. On average, 45.6 pieces of content (blogs, files and tags) were contributed each day, and by the end of testing each person had, on average, contributed 83.7 pieces of content. The quantity and quality of this analytical data was satisfactory for analysing the influence of the Hyperlinked Practice principles.

8.1.2. Questionnaire submissions

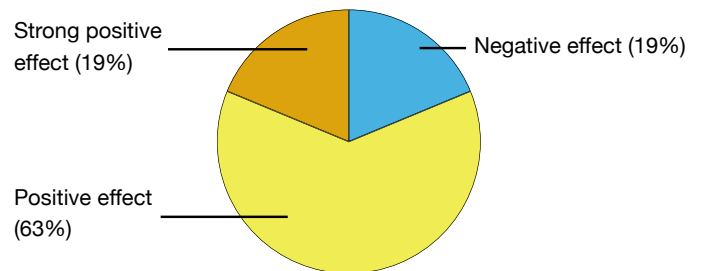
In the final weeks of testing, students were asked to complete a questionnaire on their Reasonate use and its perceived collaboration influence. Of the fifty-two students who remained in the course, thirty-two completed the questionnaire. This relatively high response rate of 61.5% increased the likelihood that the questionnaire results were representative of the students involved. However, the voluntary nature of the questionnaire, coupled with its single application at the end of testing, potentially could have led to selection bias, as those who had a better experience of Reasonate, and the course as a whole, may have been more likely to respond. Consequently, there is a risk that less favourable reviews of Reasonate may not have been captured, because those with a negative opinion may have been more likely to choose not to complete the questionnaire, or may have withdrawn earlier from the course. Fortunately, the high response rate makes this less likely to have affected the overall results.

8.1.3. The overall influence of Reasonate

Reasonate was considered a positive influence within the majority of student teams, and the BBS303 ‘Digital Craft’ course as a whole. The quantity of content published during testing is a strong indication that its core functionality was embraced by most users. Despite the workload of students and the evolving nature of the prototype, 82% of students were of the opinion that Reasonate had a positive influence on the overall course (as illustrated in Figure 8.1).

Figure 8.1: The influence of Reasonate on the student’s experience of the course

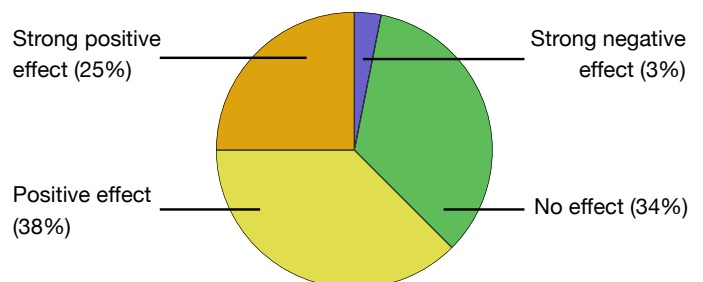
Question 14: How did Reasonate effect your experience of the BBS303 course as a whole?



In addition to this positive impression, 63% of questionnaire respondents felt that Reasonate had a positive influence on their ability to collaborate effectively within teams. As illustrated in Figure 8.2, only one respondent felt that Reasonate impaired their team’s ability to work effectively, and the remaining 34% felt it had no effect.

Figure 8.2: The influence of Reasonate on the team’s ability to work effectively

Question 15: In your opinion what effect did Reasonate have on your team's ability to work effectively?



8.2. The influence of the Hyperlinked Practice principles

The abundant data collected during the testing of the Reasonate software prototype created a strong platform for demonstrating the influence of the Hyperlinked Practice principles. The influence of the principles was demonstrated using the tests previously identified in Section 7.2. The primary focus of this analysis was to identify whether the measured effects of the principles reflected what had been inferred during the design of the prototype. If these intended collaboration effects could be achieved by applying the principles of Hyperlinked Practice, the collaboration tool will be more able to record, organise and convey a project's Building Story.

8.2.1. An overview of each principle's demonstrated influence

Overall the principles of Hyperlinked Practice achieved the majority of their desired collaboration effects during the testing of Reasonate. The outcomes of this process are summarised in Table 8.1, and described in detail within subsequent sections. The principles of ubiquity, comprehension, emulative modularity and decentralisation were demonstrated to have had a strong influence during testing. The functionality derived from the principles of situational awareness, context sensitivity and emotive semantics were not as influential, but many positive outcomes were recorded.

Table 8.1: A summary of each principle's demonstrated influence within Reasonate

Principle	Summary of demonstrated influence
Situational awareness <i>Refer to Section 8.2.2</i>	<ul style="list-style-type: none">✓ Students contributed a steady stream of updates.✓ Most students reviewed these contributions.✓ The acts of reviewing and contributing to Reasonate provided motivation for many students.✗ Students did not use Reasonate's RSS and search functions.
Ubiquity <i>Refer to Section 8.2.3</i>	<ul style="list-style-type: none">✓ The majority of students interacted with Reasonate from multiple locations during testing.✓ Students primarily used standard digital image formats to help communicate progress and decisions.

Principle	Summary of demonstrated influence
<p>Comprehension <i>Refer to Section 8.2.4</i></p>	<ul style="list-style-type: none"> • Most students were experienced Internet users, but prior to the course few had published content to the Web. ✓ The blogging and digital file upload functionality was adopted within three weeks, which was relatively fast. ✓ Students consistently used the blogging functionality. ✗ Students were slow to adopt the tagging functionality. ✗ Tagging went relatively unused until the end of the course.
<p>Context sensitivity <i>Refer to Section 8.2.5</i></p>	<ul style="list-style-type: none"> ✓ Students frequently contributed large numbers of blog posts and digital files that were sensitive to the context of their respective project teams. ✗ Contributions to the team generally occurred in clusters. ✗ The majority of teams made very limited use of tags.
<p>Emulative modularity <i>Refer to Section 8.2.6</i></p>	<ul style="list-style-type: none"> ✓ The overall number of blog posts and digital files published to Reasonate and Blogger were comparable. ✓ 2.7 times as many tags were created using Blogger, but the relative adoption by individual students was similar. ✗ Blog posts were more frequently published to Reasonate, but the majority were less than 100 words. In comparison Blogger users contributed fewer posts of longer length.
<p>Emotive semantics <i>Refer to Section 8.2.7</i></p>	<ul style="list-style-type: none"> ✓ The few users that adopted tagging used it quite extensively. ✓ Once analysed, the tag structure formed a comprehensive overview of the course's events and issues. ✗ In general, the tagging functionality was poorly adopted. ✗ Most tags were not created until the last few weeks of testing. ✗ Reasonate's user interface led to many tag synonyms.
<p>Decentralisation <i>Refer to Section 8.2.8</i></p>	<ul style="list-style-type: none"> ✓ Most students accessed Reasonate from two or more locations. ✓ Students utilised Reasonate's blog and Web Areas differently. ✓ Most of the student's personal websites had a strong preference towards the communication of digital files. ✓ The textual content was similar, but there was no relationship between the number of blog posts a student published and the size of their website. ✗ Most students used a relatively small number of hyperlinks to connect content within their blog and Web Area.

8.2.2. The effect of situational awareness

The inferred collaboration influence of situational awareness was that students would be more informed of the progress of team members and the class as a whole. Reasonate embodied this principle through the inclusion of RSS news-feeds, email notifications and search functionality. The intention of this functionality was to promote regular and frequent contributions by students, and to encourage timely reviews of the work undertaken by team members. This principle was therefore tested by assessing three usage characteristics:

- The frequency and regularity of content contributions to Reasonate.
- The use of Reasonate’s integrated RSS, email and search functionality.
- The tendency for students to use Reasonate to review content.

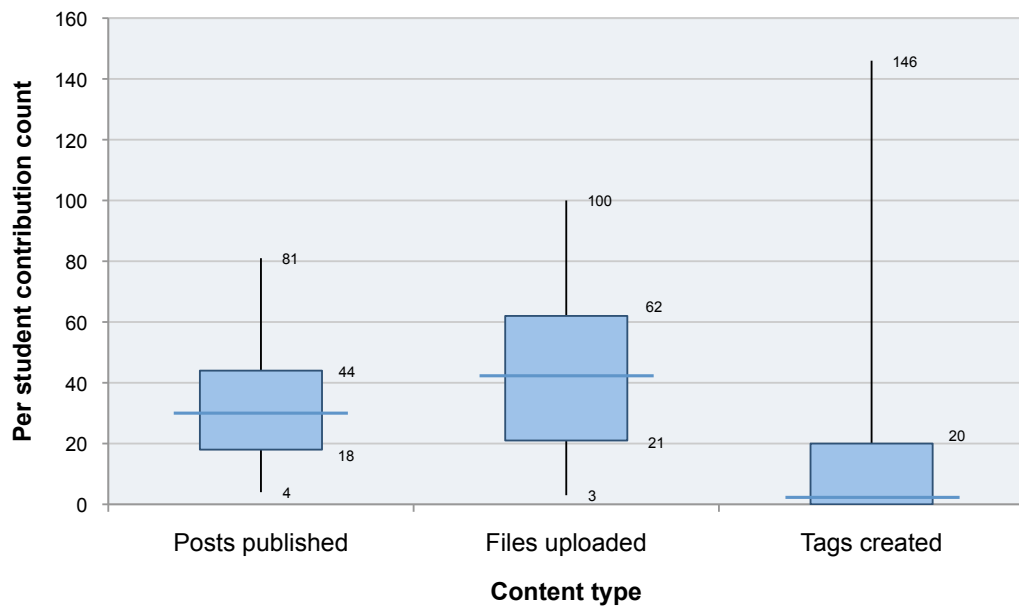
8.2.2.i. Frequent contributions by the majority of participants

Overall, Reasonate users regularly contributed blog posts and digital files throughout the testing process. As detailed in Table 8.2, the most prolific students contributed five times as many pieces of content per week as students in the lower quartile. Figure 8.3 illustrates that over the duration of testing, this led to a wide range in the total amount of content contributed by students. Twenty-two students published a new post twice a week or more on average, and twenty published approximately one post per week. The remaining ten averaged less than one post per week, which was lower than expected given the course’s workload and timeframe.

Table 8.2: An analysis of the content published per student in an average week

	Posts published in an average week	Files uploaded in an average week	Tags created in an average week
Maximum	5.06	6.25	9.13
Upper quartile	2.75	3.88	1.25
Average	2.07	2.68	1.01
Median	1.88	2.63	0.22
Lower quartile	1.13	1.31	0.00

Figure 8.3: A box-plot analysis of the total Reasonate contributions per student



These contributions to Reasonate took place throughout the day and during the night. As illustrated in Figure 8.5, this steady usage pattern was established within the first few weeks of testing, and remained relatively constant.

8.2.2.ii. No RSS or search uptake amongst students

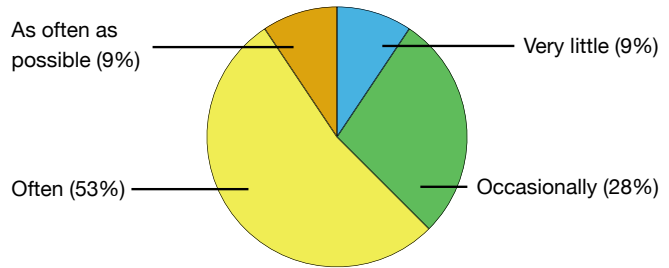
None of the students who responded to the questionnaire had experimented with, or consistently used, Reasonate's RSS or search functionality to monitor new content.

8.2.2.iii. Reasonate promoted regular reviews of work and improved motivation

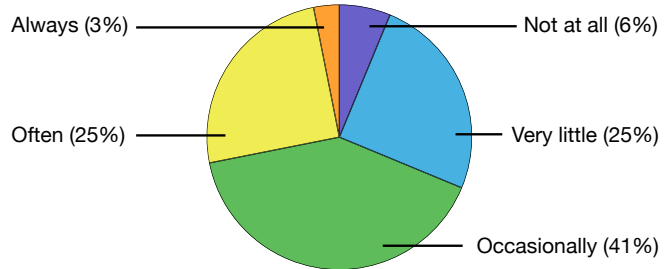
Reasonate was successful at promoting student reflection and situational awareness during the testing period. As illustrated in Figure 8.4, 62% of questionnaire respondents indicated that they often used Reasonate to review their project and the work of others. Although only 28% of respondents reported that documenting their progress often or always encouraged further critical thinking, 53% of respondents found that the process of reading and writing the blogs on Reasonate provided extra motivation during the course.

Figure 8.4: Reasonate's influence on the reflection process and student motivation

Question 4: How frequently did you use Reasonate to review your project or other people's work?



Question 8: Did the act of documenting your progress on Reasonate encourage further critical thinking on the work you had undertaken?



Question 9: Did the act of reading and writing these blogs provide extra motivation in the course?

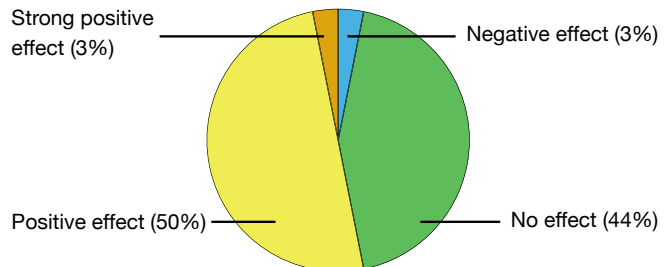
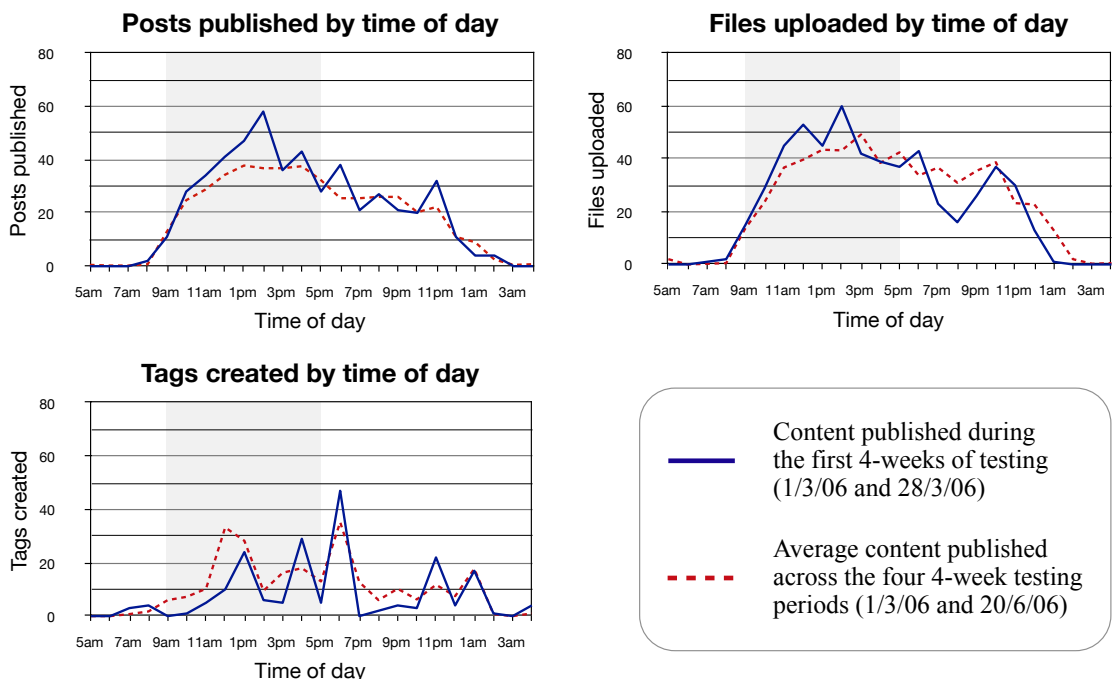


Figure 8.5: Activity graphs comparing the times of day contributions occurred



8.2.2.iv. Summary

The demonstrated collaboration influence of situational awareness within Reasonate was mixed. Overall, the tool garnered a steady stream of updates from the majority of users throughout the day. Most students regularly reviewed this content, and more than half reported that reading and writing the blogs provided extra motivation in the course. Unfortunately, students did not use Reasonate's RSS and search functionality to monitor new content contributions.

8.2.3. The effect of ubiquity

The inferred collaboration influence of ubiquity was that students would be more able to participate in digital design conversations when the enabling technologies are readily available and well understood. Reasonate embodied this principle in its web-based design, which enabled users to access the service from any standards-based web browser. The intention of this design decision was to ensure students could frequently interact with the prototype from a variety of locations. It was inferred that this accessibility and flexibility would encourage users to exchange information in ubiquitous formats that others could easily access without specialised software. This principle was therefore tested by assessing three usage characteristics:

- The frequency with which users interacted with Reasonate throughout testing.
- The tendency of users to access Reasonate from different locations.
- The tendency of users to communicate using ubiquitous digital file formats.

8.2.3.i. Frequent interactions from a number of different locations

The majority of students reviewed and contributed Reasonate content from two or more locations. As illustrated in Figure 8.6, 66% of respondents used the prototype from their place of residence, and 13% from a professional workplace. This flexibility promoted frequent interactions, with 44% of questionnaire respondents of the opinion that they often contributed to Reasonate (see Figure 8.7). This finding was supported by similar results in the analytical data described in Section 8.2.2.i and Table 8.2.

Figure 8.6: The general locations where students interacted with Reasonate

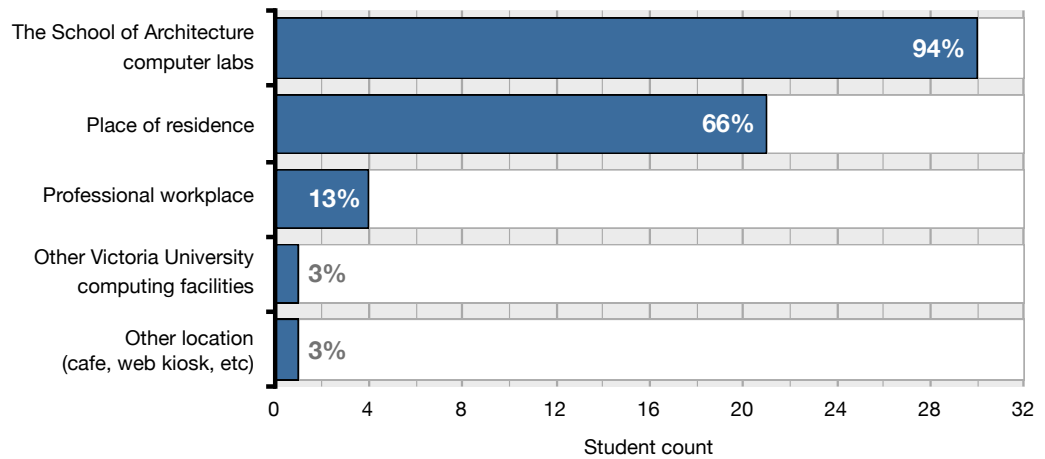
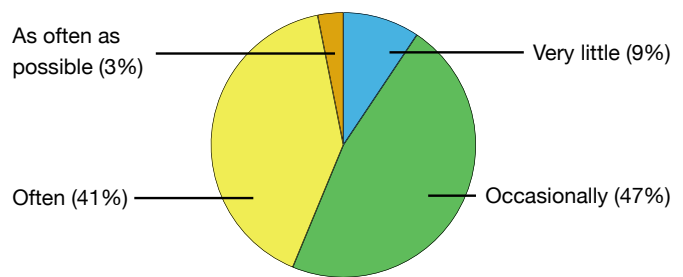


Figure 8.7: The student’s perception of their blogging frequency during testing

Question 6: In your own opinion how frequently did you post new blog entries on Reasonate?



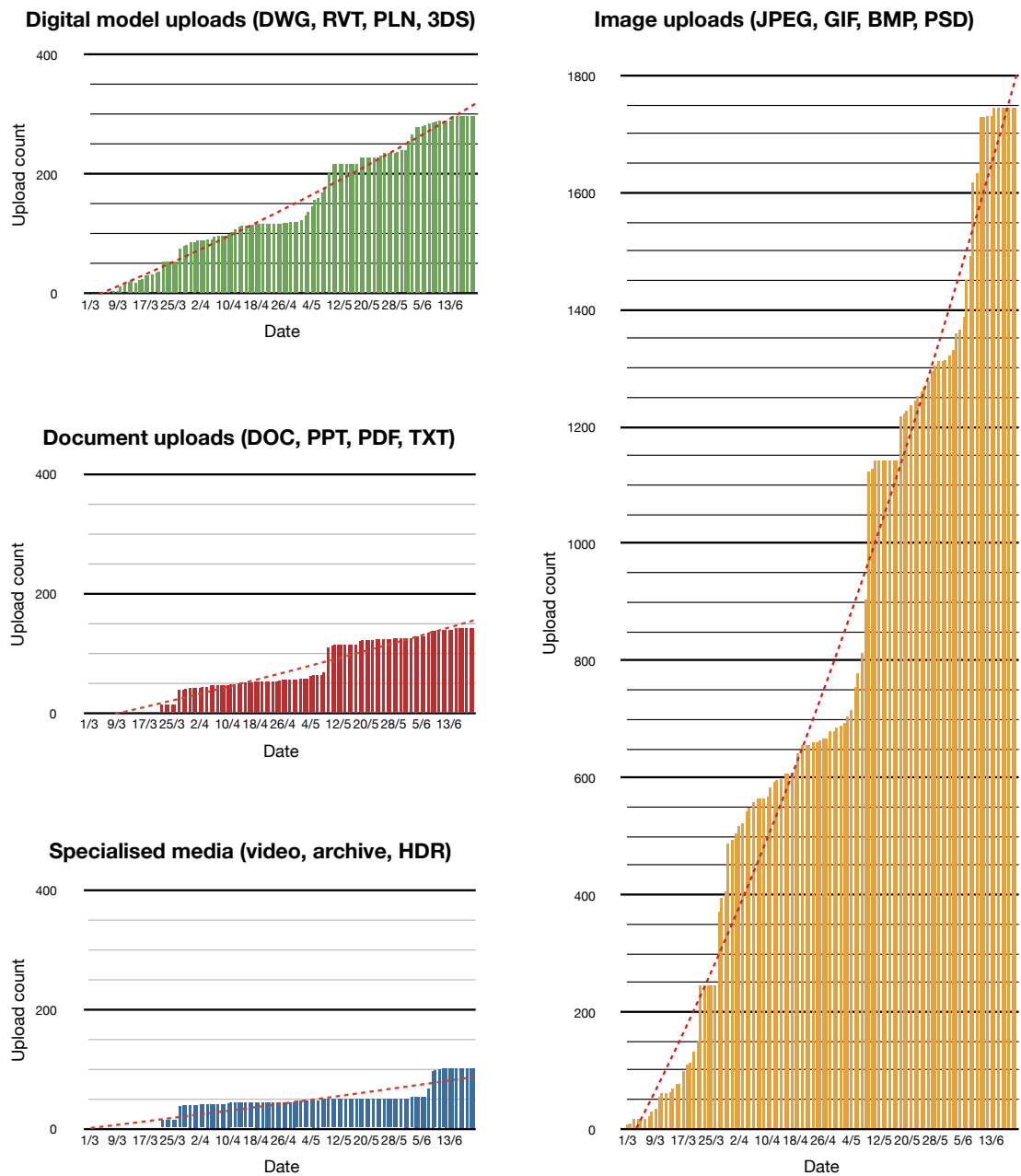
8.2.3.ii. Images supported by Reasonate blog posts as the dominant form of communication

Users uploaded 2,275 digital files to Reasonate, and of those 1,745 (76.7%) were images (see Table 8.3). Only 48 of these images were in formats that could not be viewed natively within a modern web browser. Digital images were frequently uploaded to Reasonate in large numbers, as illustrated in Figure 8.8. In comparison, most other digital formats underwent small and sporadic growth, that often coincided with coursework hand-ins.

Table 8.3: A breakdown of the digital files uploaded to Reasonate by format

Digital Model				Images				Documents				Specialised Media			
AutoCAD	Revit	ArchiCAD	3D Studio	JPEG	GIF	Non-web Image	Photoshop	PDF	Word	Powerpoint	Plain Text	Movie (AVI / MOV)	High Dynamic Range	Archive (ZIP / RAR)	Google Earth
124	81	62	29	1677	20	40	8	6	5	121	9	54	35	9	3

Figure 8.8: A breakdown of the digital file formats uploaded to Reasonate over time



8.2.3.iii. Summary

The demonstrated collaboration influence of ubiquity within Reasonate was strong. The majority of students accessed the web-based prototype from locations other than the School of Architecture. This capability supported more frequent interactions with Reasonate. When these contributions occurred, students typically used standard digital image formats to illustrate their progress and support their blog posts. This decision ensured that the information conveyed could be reliably accessed by any interested party with an Internet connection and modern web browser.

8.2.4. The effect of comprehension

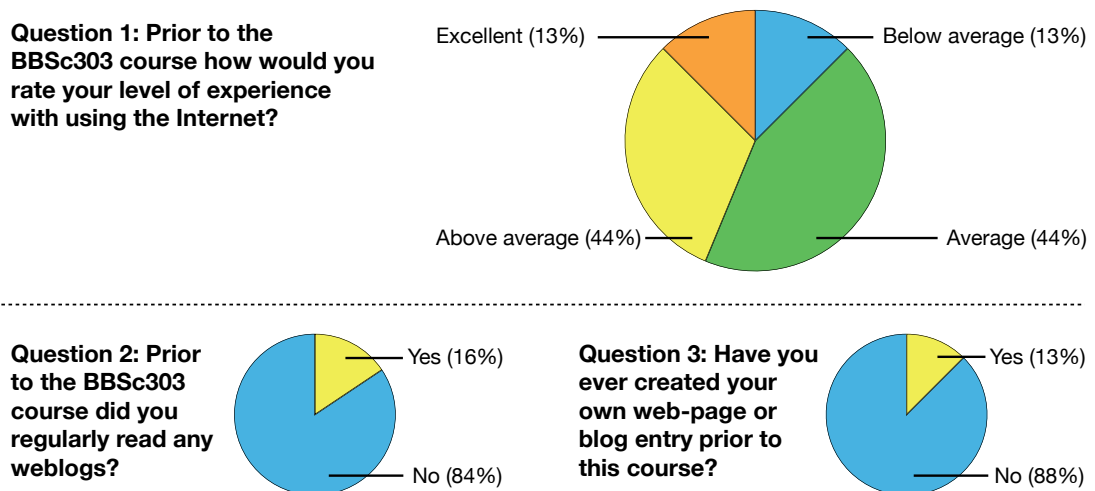
The inferred collaboration influence of comprehension was that students would be able to easily learn and leverage a digital tool that had a clear purpose, straightforward processes, and a cohesive user experience. Reasonate embodied this principle in its core functionality, which was concise, clearly presented, and deployed in an incremental manner that was intended to lessen the students' learning curve. The intention of these design and deployment decisions were to ensure that irrespective of the user's prior experience, they would be capable of quickly adopting the tool and consistently using it throughout the course. This principle was therefore tested by assessing the students' experience using the Internet, and two usage characteristics:

- The rate users adopted Reasonate's core functionality.
- The consistency that Reasonate's functionality was used during the course.

8.2.4.i. An experienced user group with little background in Web content publishing

57% of respondents considered themselves as having an above average or excellent level of experience when it came to using the Internet (see Figure 8.9). No student who responded felt that they had a poor understanding of the Internet, but four respondents (13%) considered their experience to be below average. Although the perceived experience of the testing group was high, only 16% regularly read weblogs, and 13% had published a webpage or blog prior to taking part in the course.

Figure 8.9: The students' Internet experience prior to the course



8.2.4.ii. A strong level of adoption, but inconsistent use of tagging

Reasonate’s blogging functionality was adopted by the majority of students within three weeks. However, aside from bursts at the beginning and end of testing, the tagging functionality struggled to gain traction. Students had little content to publish during the first week of testing, but by the third week the daily number of published posts had reached a strong and relatively consistent level of approximately 16 per day (see Figure 8.10). The tagging functionality was tested by students when it was introduced, but as illustrated in Figure 8.11, it did not achieve the same level of consistent use.

Figure 8.10: The daily growth and running total of posts published to Reasonate

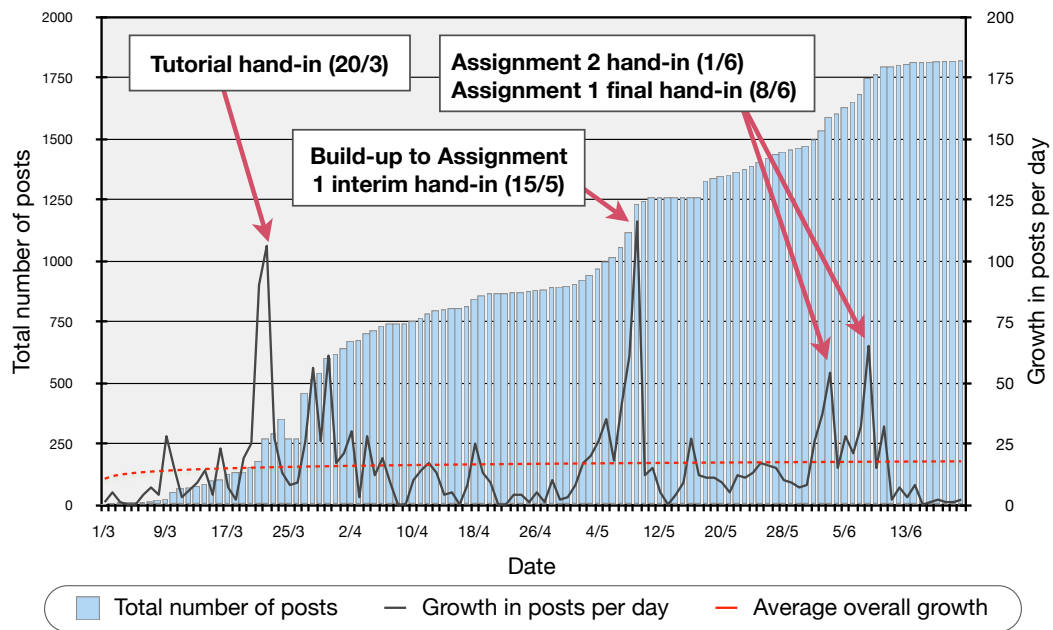
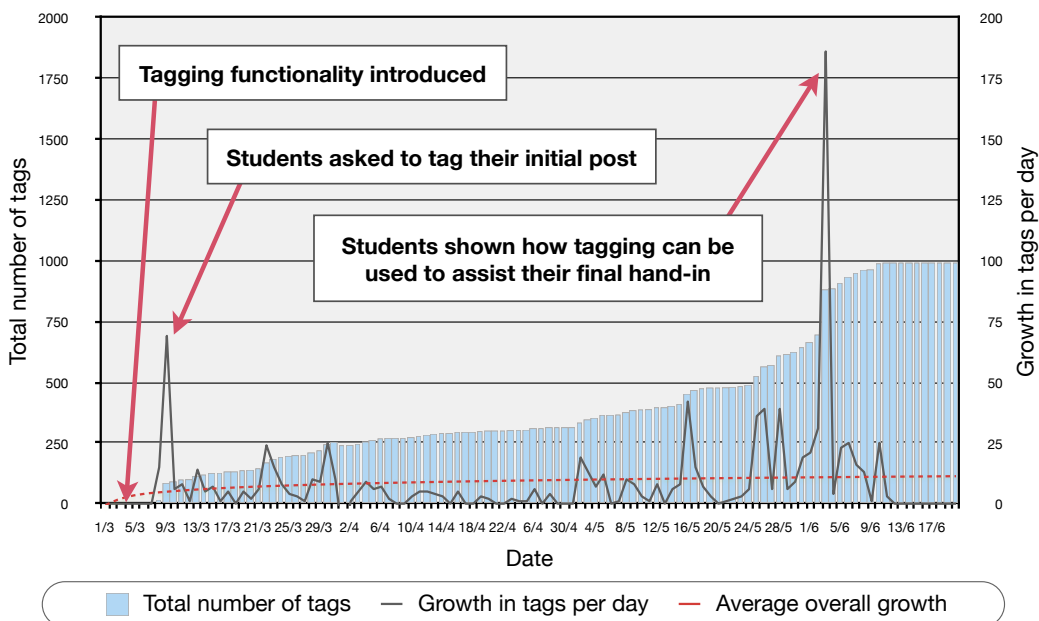


Figure 8.11: The daily growth and running total of tags created within Reasonate



8.2.4.iii. The consistent use of Reasonate's blogging and file uploading functionality

The blogging and digital file uploading functionality of Reasonate was used consistently throughout testing, but tagging went relatively unused until the end of the course. As described in Section 8.2.2.i and illustrated in Figures 8.8 and 8.10, the majority of users consistently contributed blog posts and digital files. In contrast, the majority of users made little or no use of tags (see Figure 8.3), and those that did used the functionality inconsistently. As illustrated in Figure 8.11, the majority of tags were not applied until students were organising their coursework for final submission in the final weeks of testing.

8.2.4.iv. Summary

The demonstrated collaboration influence of comprehension was generally strong, but not all aspects of Reasonate were well understood or used. Most students considered themselves to be relatively experienced in using the Internet, but only a few had published blogs or webpages to it. Despite this, within three weeks the majority of the class were regularly publishing blog posts and digital files to Reasonate. Contributions reached a relatively high level early in the testing process, and remained strong for the duration. However, there was only limited adoption of Reasonate's tagging functionality.

8.2.5. The effect of context sensitivity

The inferred collaboration influence of context sensitivity was that students would be more able to understand events and associated information when it is presented in a manner that is relevant to the current collaboration context. Reasonate embodied this principle in its ability to aggregate and present contributions from team members as a simple, project-specific blog. In addition, team members could easily browse the semantic structure formed when they and fellow members of the team tagged project content. The intention of these project-specific vocabularies was to improve the understanding of students, by highlighting the current issues and topics of interest within the team.

To test this principle three usage characteristics were assessed:

- The quantity of the project specific contributions made to the team by its members.
- The frequency and regularity of the contributions made by team members.
- The use of tags by team members to organise project information.

8.2.5.i. A wide range in the quantity of content contributions to project teams

The project teams formed within Reasonate exhibited the same varied contribution patterns that were observed at the individual student level (see Figure 8.3). When the total content contributions to projects were analysed, as illustrated in Figure 8.12, there were some extreme results, and the interquartile range for all forms of content was relatively broad. Of the twenty-four project teams, seventeen had over fifty project specific blog posts and digital files contributed to them (see Figure 8.14), which over the twelve week assignment period was a reasonable performance. Within most teams there was a relatively even balance of blog posts to digital files. However, where this balance did not exist (teams 33, 49, 9 and 8), there was a very strong tendency towards collaborating and recording progress with digital files instead of blog posts.

Figure 8.12: A box-plot analysis of the total Reasonate contributions per project

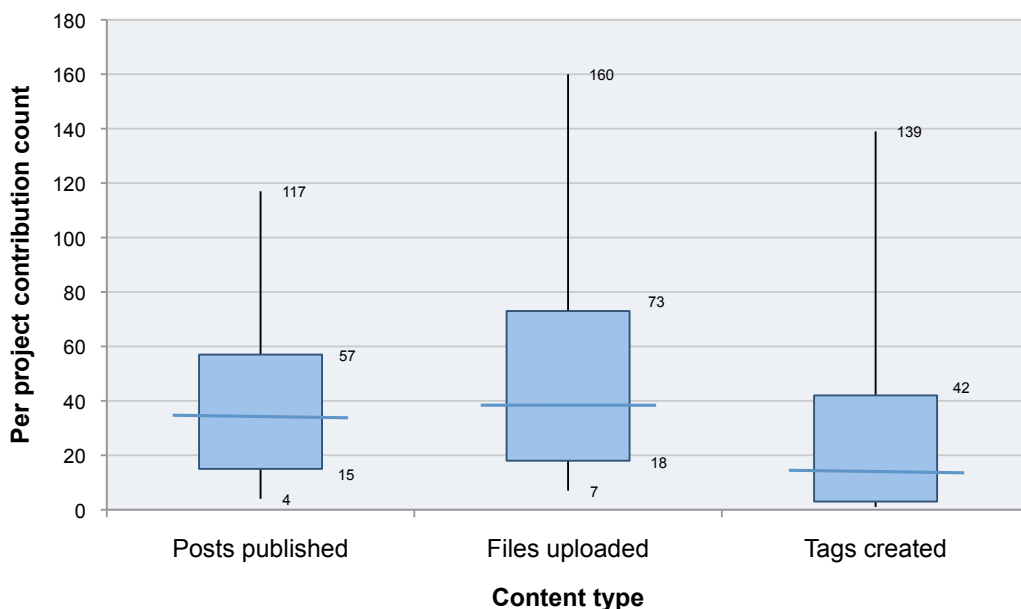
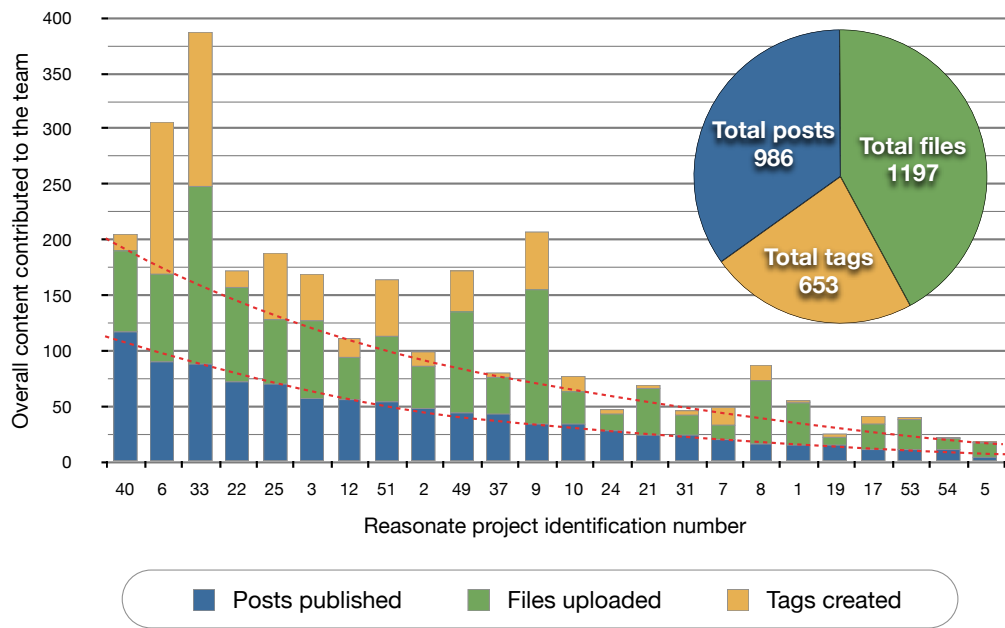


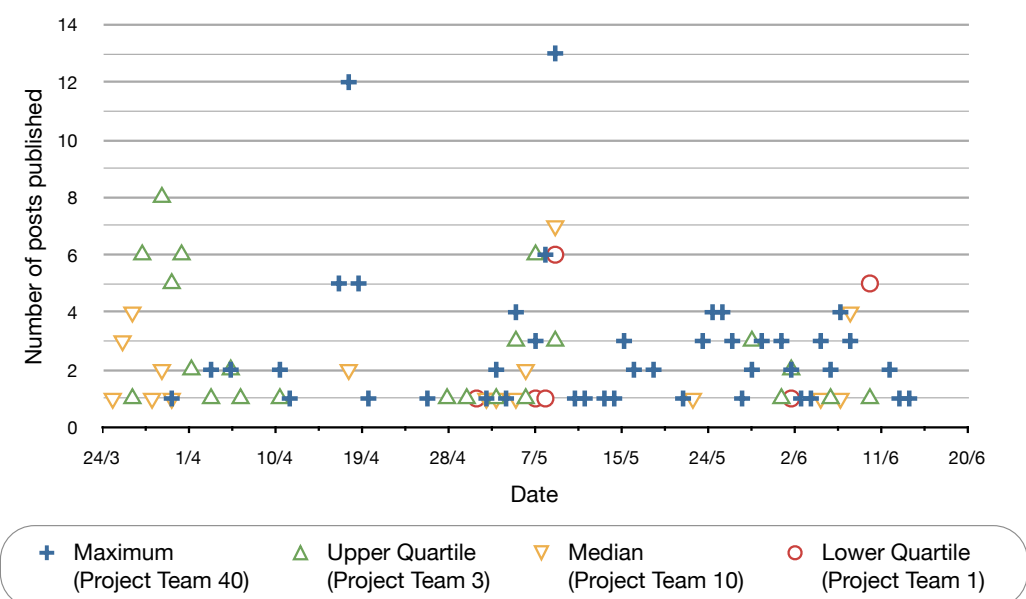
Figure 8.13: The breakdown of content contributed to each project ordered by the total number of blog posts



8.2.5.ii. Frequent but often inconsistent contributions

The majority of the students frequently contributed content to their project team, but often this activity occurred in clusters. As illustrated in Figure 8.14, teams that published the most information on their progress, for example Project Team 40, did so by making frequent and consistent contributions throughout the course. In contrast, teams who published less content, for example Project Team 1, made frequent contributions, but generally only around the time of a submission deadline.

Figure 8.14: The content contribution activity of four representative project teams



8.2.5.iii. Low adoption of tagging by the majority of teams

The majority of project teams did not use Reasonate's tagging functionality frequently or consistently. However, compared to the usage patterns of individuals illustrated Figure 8.3, project teams made greater use of tags (see Figure 8.12). There was a median of fourteen project tags and a lower quartile of three, which reflected the inconsistent manner tagging was applied within different teams. Figure 8.13 is an illustration of this inconsistent adoption, especially among teams who contributed a considerable number of blogs and digital files, such as Project Teams 40, 22 and 12.

8.2.5.iv. Summary

The demonstrated collaboration influence of context sensitivity was mixed. A sizeable number of project specific posts and digital files were contributed by the majority of project teams. These contributions typically occurred in clusters of activity that were centred around assignment deadlines. Unlike the relatively high number of blog and digital file contributions, most teams made minimal use of Reasonate's tagging functionality to help organise and navigate project content.

8.2.6. The effect of emulative modularity

The inferred collaboration effect of emulative modularity was that all or part of a digital collaboration tool could be replaced by a third-party system which emulated the original's modular design and functionality. Reasonate embodied this principle in its underlying design, which centred around four components:

- The ability to publish simple HTML-based messages.
- The ability to upload any type of digital file and associate it with a blog post.
- The ability to apply semantic tags to blog posts and digital files.
- The ability for users to upload a custom website that summarised the project.

In 2007 these four pieces of functionality were independently emulated within the same BSc303 'Digital Craft' course using a combination of Google Blogger, a Google Group, and a Victoria University FTP server. The demonstration of emulative modularity was whether the fundamental collaboration process and dynamic that were fostered by Reasonate could be replicated using this independently created system.

The principle was therefore tested by measuring three usage characteristics from the testing of the Reasonate (2006) and Blogger (2007) collaboration tools:

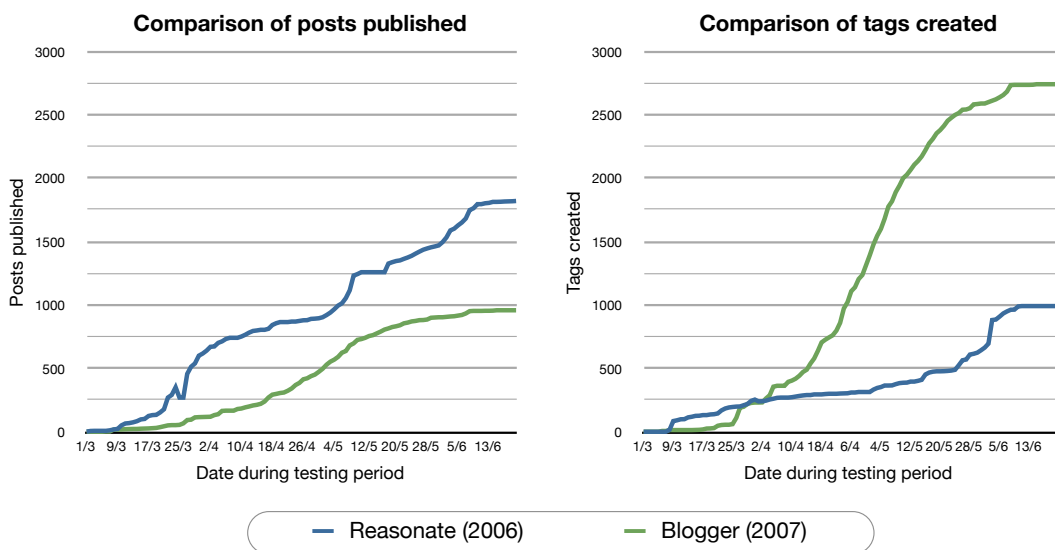
- The rate by which users contributed different forms of content to both systems.
- The quantity of the content contributed by users of both systems.
- The average length of the blog posts published to both systems.

8.2.6.i. Similar and contrasting patterns of content growth

The adoption rate of the Blogger-based system was considerably slower for blog posts and tagging, as illustrated in Figure 8.15. However, by the 1st April users of both systems were contributing a similar overall number of blog posts. Nevertheless, while overall growth was similar, the majority of Reasonate's posts were contributed during short, intense periods of activity, whereas Blogger witnessed steady and consistent growth for the remainder of testing.

Tagging in both systems took a long time to gain adoption, but the rate of Blogger tag growth eclipsed that of Reasonate after five weeks of testing. This high rate of tag growth continued unabated for the majority of the course, whilst Reasonate only achieved a similar level of growth in the final weeks of testing.

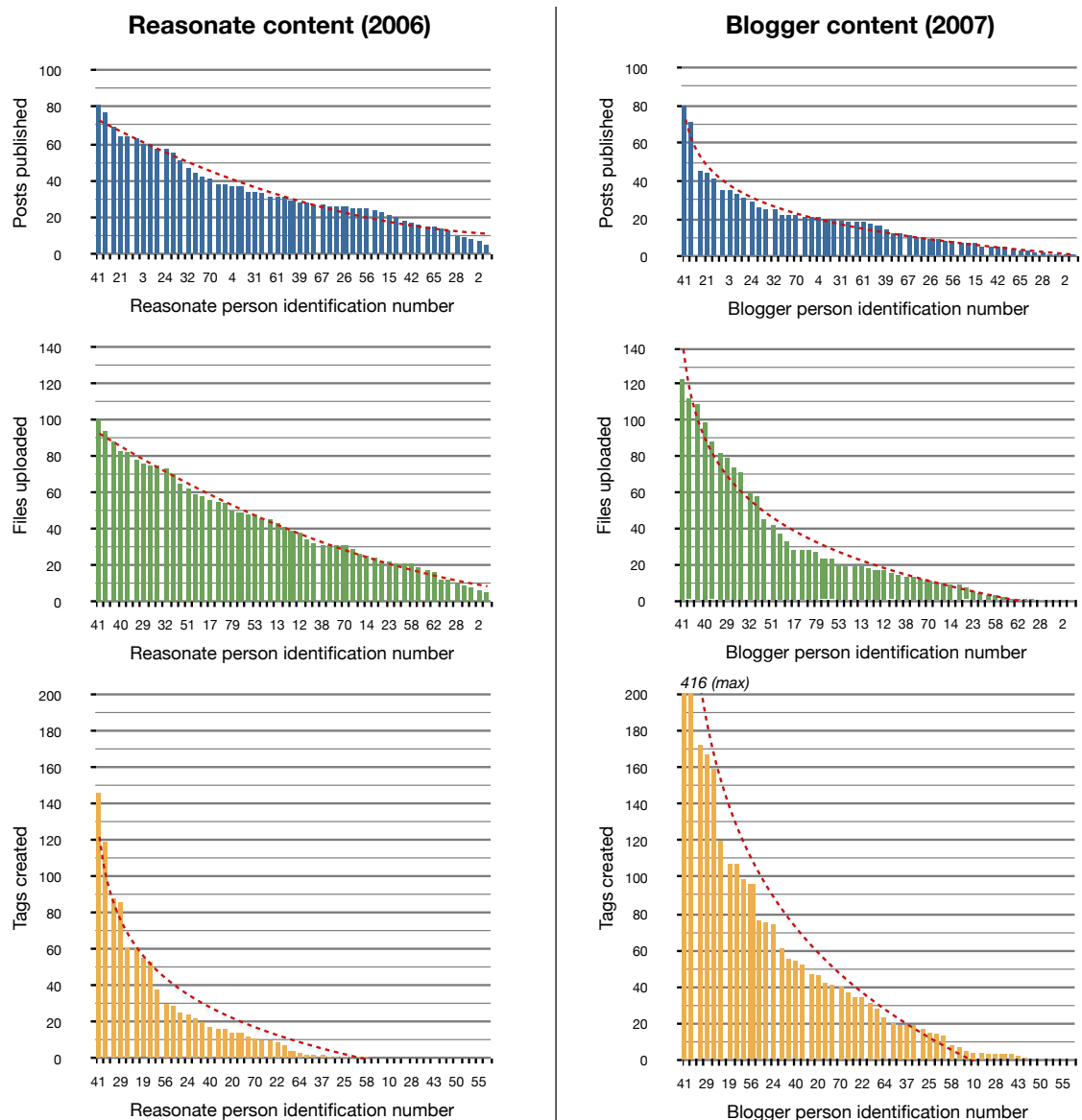
Figure 8.15: A comparison of the overall post and tag growth rates during the Reasonate (2006) and Blogger (2007) testing periods



8.2.6.ii. Similarities in individual contributions of posts and digital files, overshadowed by tags

Both systems shared similarities when it came to the wide range in the total number of blog posts and digital files that were contributed by individual students. The blog post and digital file contributions of Reasonate users were distributed in a relatively linear pattern, as illustrated in Figure 8.16. In comparison, most Blogger users did not contribute as many blog posts or digital files, but the top contributors from each group were relatively similar. The total number of tags created by Blogger users was 2.7 times that of Reasonate. Although the total number of tags created in the two systems differed enormously, the relative distribution of the tags between users was similar, with the top 10% of users creating 49% (Reasonate) and 43% (Blogger) of all tags.

Figure 8.16: An ordered comparison of the total contributions to Reasonate and Blogger by individual users



8.2.6.iii. Large differences in the length of blog post content contributed

On average, the majority of Reasonate’s blog posts (82%) were less than 100 words in length, compared to only 45% of Blogger posts. Figure 8.17 illustrates that all but two users of Reasonate preferred to compose short blog posts, less than 100 words in length. In contrast, the blog posts published by Blogger users were evenly distributed between short and comparatively long posts, as illustrated in Figure 8.18.

Figure 8.17: The quantity and length of blog posts contributed to Reasonate (2006)

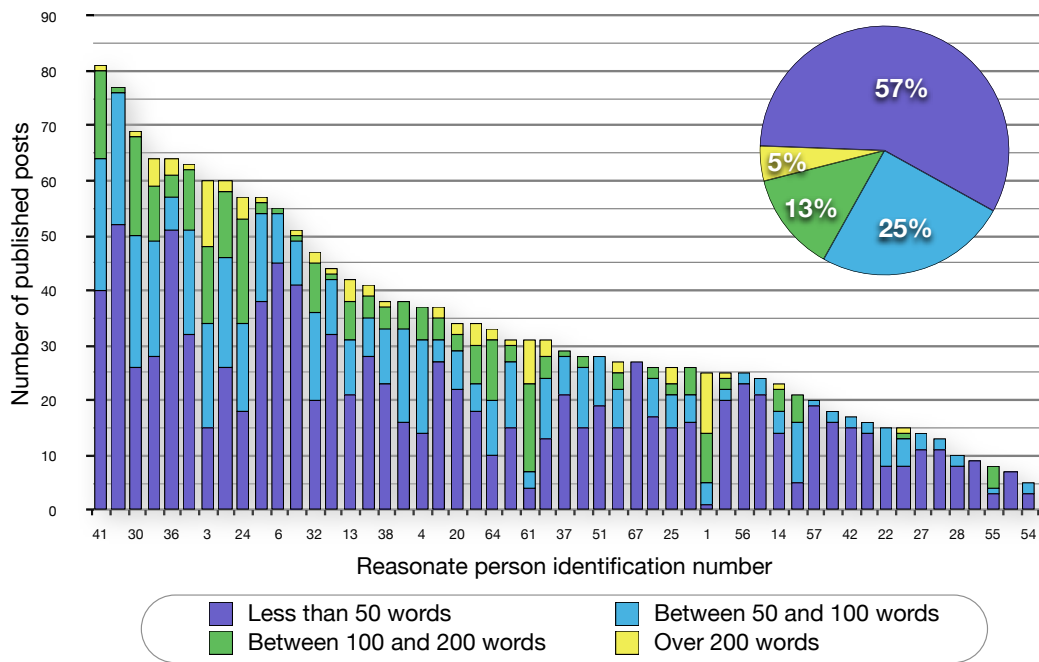
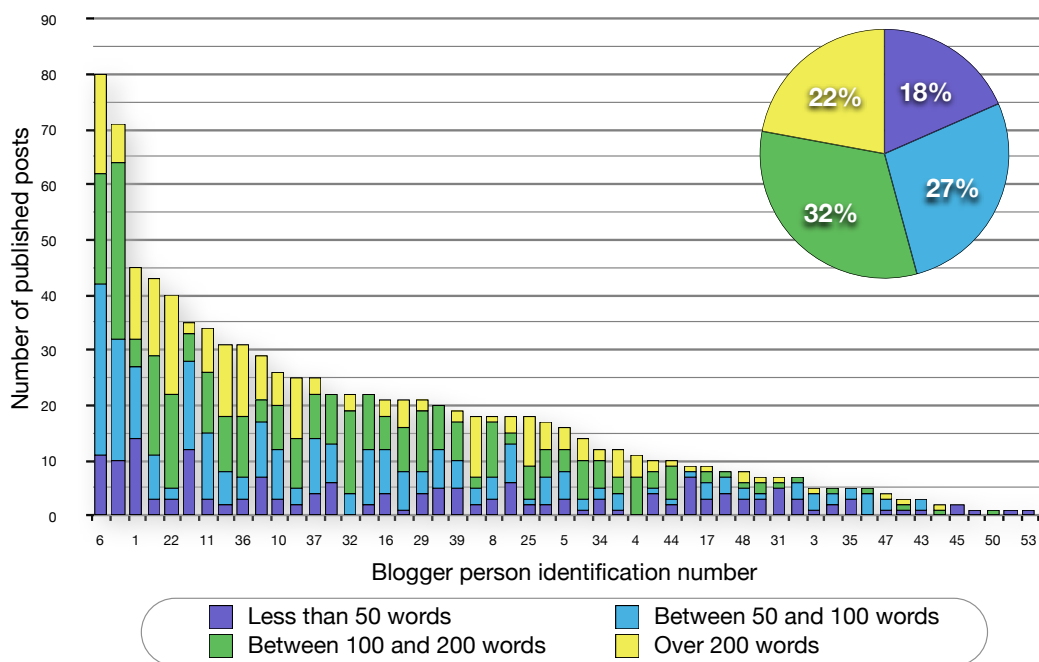


Figure 8.18: The quantity and length of blog posts contributed to Blogger (2007)



8.2.6.iv. Summary

The demonstrated collaboration influence of emulative modularity was generally strong and there was a similar number of blog post and digital file contributions to both systems. However, the Blogger system did under-perform and outperform Reasonate in certain areas. Reasonate garnered the most blog posts, but the majority were less than 100 words in length, whereas Blogger users tended to publish a broader mixture of short and long posts. The use of tagging in both systems differed greatly. There were 2.7 times as many tags were created by users of Blogger compared to Reasonate, and these were created more frequently and consistently throughout testing.

8.2.7. The effect of emotive semantics

The inferred collaboration effect of emotive semantics was that participants would generate tailored semantic structures that reflected the important events and concepts of the project. The Reasonate software prototype embodied this principle within its tagging mechanism, which allowed students to assign any number of semantic tags to blog posts and digital files. Once assigned, these tags were organised into three distinct groups, tags created by the individual, tags assigned by team members to team content, and tags assigned by any Reasonate user to any content. Users could review and navigate these different tag structures whilst reflecting upon contributed work, and they could provide an indication of evolving events and issues within the team. These principles were therefore tested by measuring three usage characteristics:

- The frequency and regularity that tags were applied by students during testing.
- The total number of tags created by each student.
- The semantic structures that were formed as a result of this tagging process.

8.2.7.i. Tags were infrequently and inconsistently applied during testing

Twenty-three percent of questionnaire respondents did not use Reasonate's tagging functionality (see Figure 8.19), a figure which is supported by the analytical data illustrated in Figure 8.11. Sixteen percent of students responded that they frequently or always tagged content, but the analytical data shows that tags were not frequently or consistently created until the final stages of testing. The reasons given for tagging content were indicative of the low adoption and usage patterns. As illustrated in Figure 8.20, 38% of respondents

experimented with the functionality after its introduction, but only 25% continued to use tagging in order to help find content.

Figure 8.19: The students' perception of their tagging habits during testing

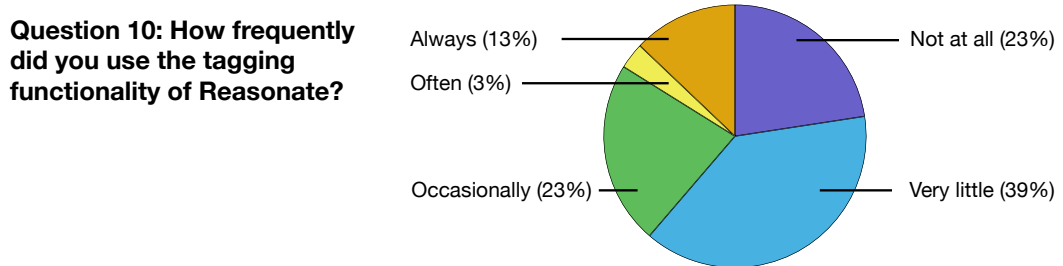
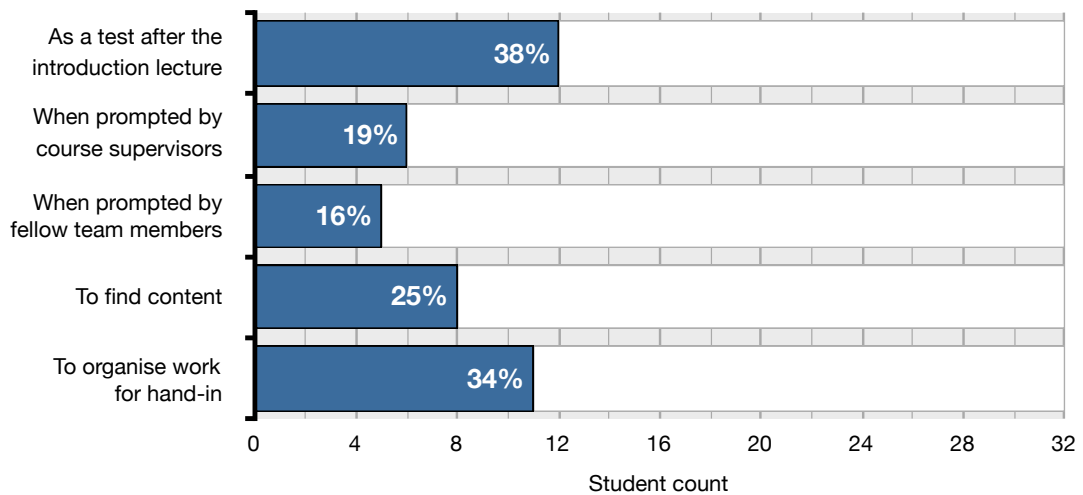


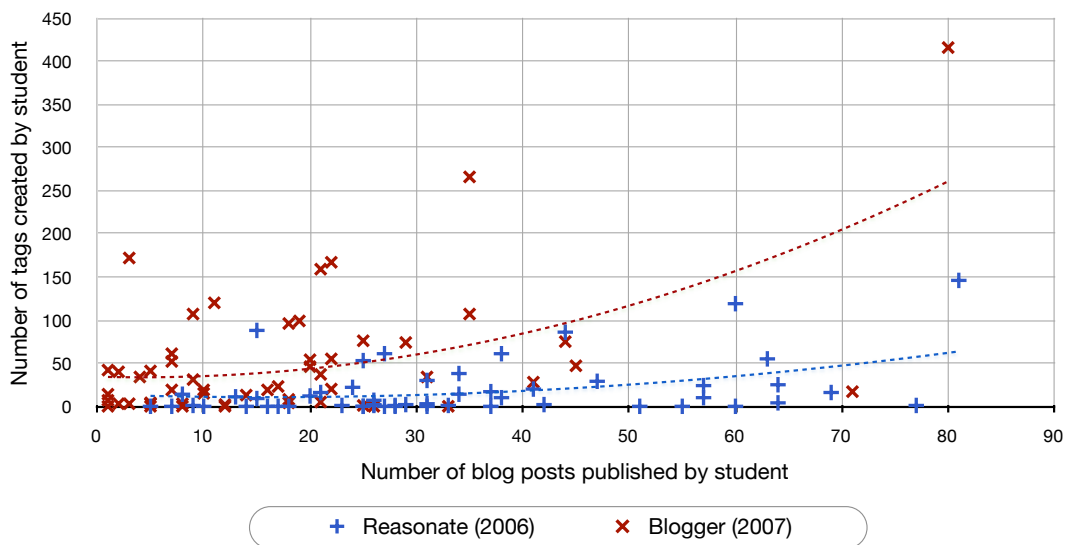
Figure 8.20: The reasons given by students for tagging Reasonate content



8.2.7.ii. Tags were only applied by a minority of users

Overall, Reasonate's tagging functionality was under-utilised by the majority of students. As illustrated in Figures 8.3 and 8.16, only nine students created more than 30 tags, which is low considering the number of blog posts and digital files contributed over the same period. However, the few users that did utilise Reasonate's tagging functionality tended to be quite prolific, for example the top five tag contributors created 500 tags. Tags were more commonly applied during group work(see Section 8.2.5.iii), but half of the groups still only created fewer than 18 tags each. As illustrated in Figure 8.21, there was very little relationship between the total number of blog posts a student published, and the number of tags they created. This result was consistent across the testing of Reasonate in 2006 and Blogger in 2007.

Figure 8.21: A comparison of the total number of blog posts and tags created per Reasonate and Blogger user



8.2.7.iii. A rich and descriptive semantic structure was formed

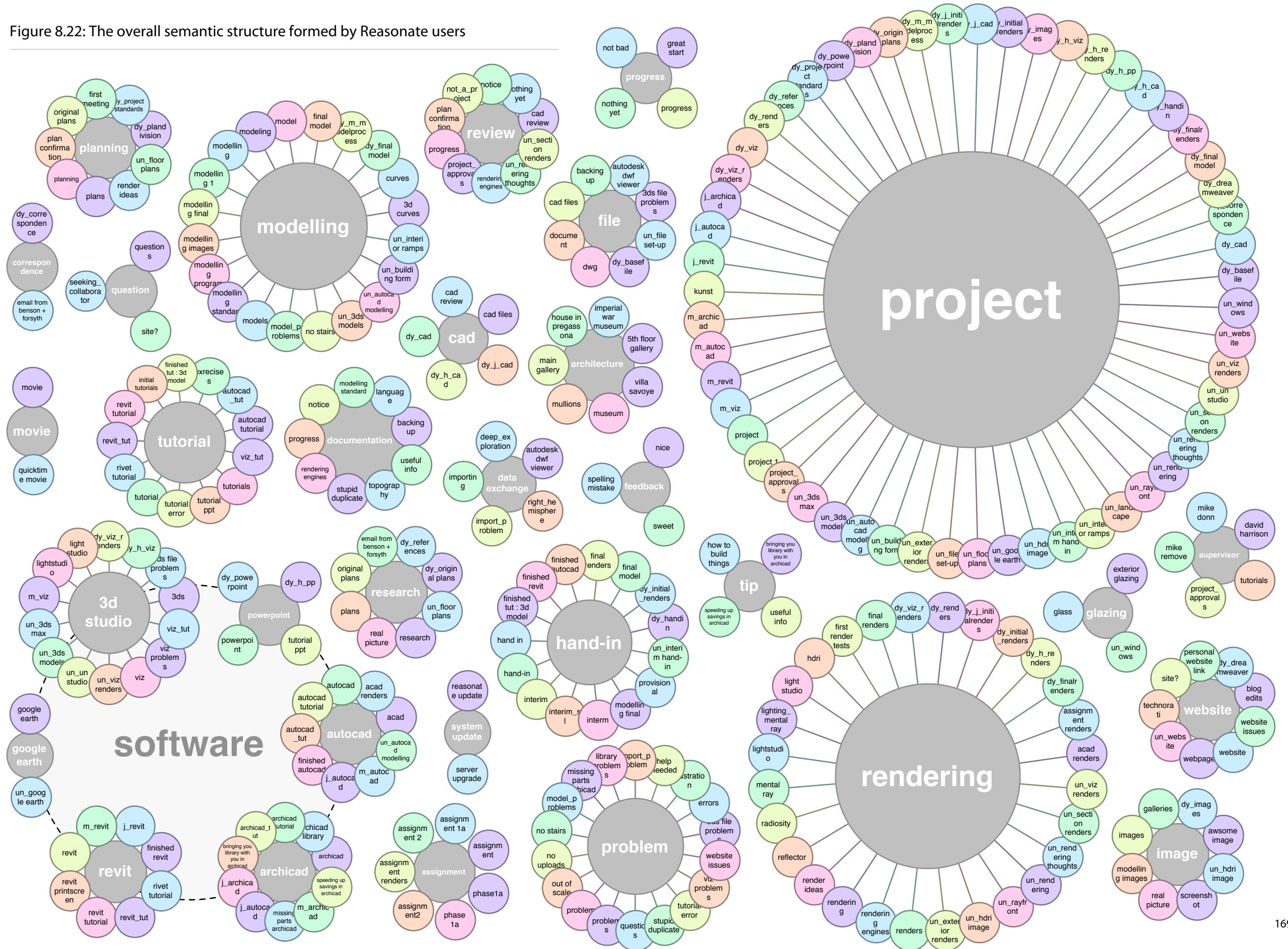
Students that did utilise Reasonate’s tagging functionality managed to create a relatively comprehensive description of their project and the course as a whole, which is illustrated in Figure 8.22. Shortcomings in Reasonate’s user interface led some teams to prefix their tags with unique identifiers, for example ‘dy_’ and ‘un_’, so that they could differentiate their tags from other projects. This shortcoming was addressed during testing (see Section 7.4.3.ii), but many teams continued to use this convention. As a result of this prefixing, spelling irregularities and synonyms, many semantic concepts were represented by multiple tags, for example ‘modeling’, ‘modelling’, ‘dy_modelling’ and ‘modelling final’. As a consequence, constructing this tag overview required manual grouping, for example:

cad - ‘cad files’, ‘cad review’, ‘dy_cad’, ‘dy_h_cad’, ‘dy_j_cad’

8.2.7.iv. Summary

It was difficult to demonstrate the collaboration influence of emotive semantics in this test of Reasonate. The majority of users made little or no use of the tagging functionality, and those who did create larger numbers of tags primarily did so during the last few weeks of testing. Although the overall use of tags was low, the semantic structures formed were comprehensive, and indicative of the content and issues experienced during the course. However, the construction of this semantic overview required a considerable amount of manual organising. This was primarily due to user interface shortcomings in Reasonate, which led to the propagation of many tag synonyms.

Figure 8.22: The overall semantic structure formed by Reasonate users



8.2.8. The effect of decentralisation

The inferred collaboration effect of decentralisation was that participants would have more opportunity and means to communicate if they were not limited to using one digital tool based in a single location. The Reasonate software prototype embodied this principle in its web-centric design that allowed students to interact with it from any Internet-connected web browser. In addition Reasonate had two separate, but related repositories for content, a tightly controlled and presented blog, and the Web Area where students could upload personal websites of their own design. To test the influence of this principle, three usage characteristics were measured:

- The locations where students chose to interact with Reasonate from.
- The composition of the student's free form Web Area, compared to the content they had contributed to Reasonate's structured blog.
- The student's use of hyperlinks to weave together the information they contributed within blog posts and personal websites into a cohesive story.

8.2.8.i. Regular interactions from a number of locations

Besides the School of Architecture computer labs, 66% of questionnaire respondents interacted with Reasonate from their place of residence (see Figure 8.6). Some students did access the tool from other locations, but this was not nearly as prevalent.

8.2.8.ii. Distribution fostered diverse communication strategies

The student websites published to Reasonate's Web Area used a wide range of aesthetic, organisational and communication strategies, a selection of which are illustrated in Figure 8.23. A high portion (47%) of the websites published to the Web Area had over 100 digital files, excluding HTML web pages. As illustrated in Figure 8.24, there was no overall relationship between the quantity of digital files contributed to a student's blog and Web Area. The difference between the written content contributed to both sections of Reasonate was less marked, with each student contributing on average 35.3 blog posts and 28.2 HTML files. Fifty-six percent of students published more blogs to Reasonate than uploaded HTML files to the Web Area.

Figure 8.23: Examples of student websites published to Reasonate's Web Area

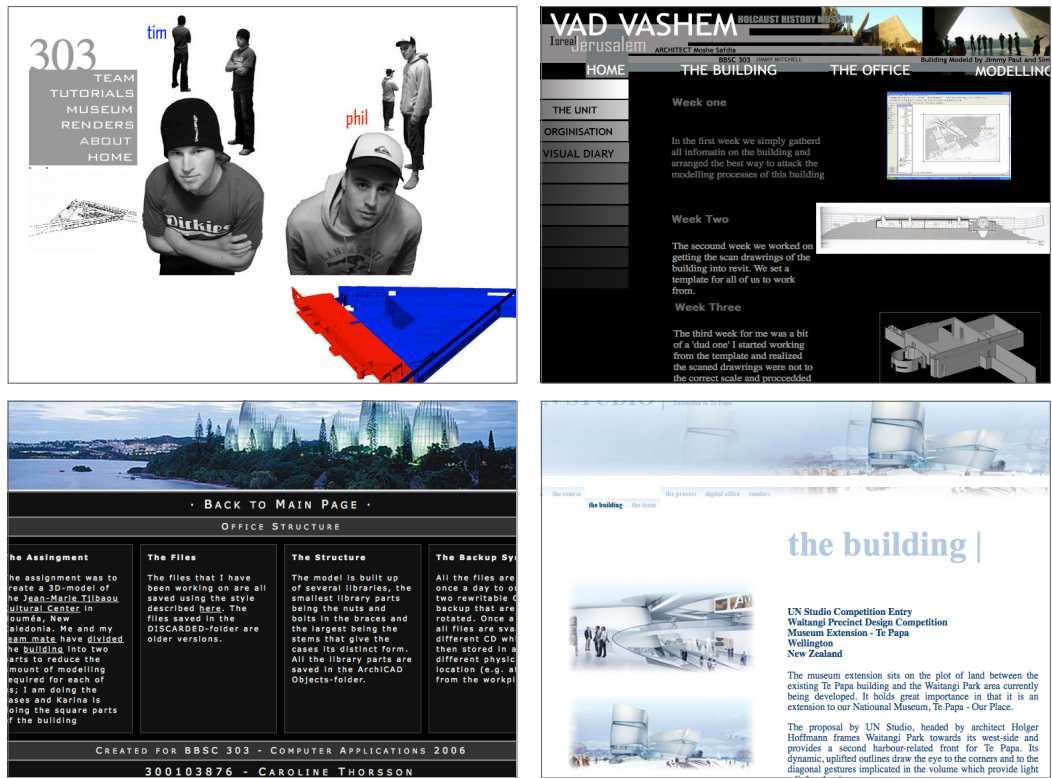
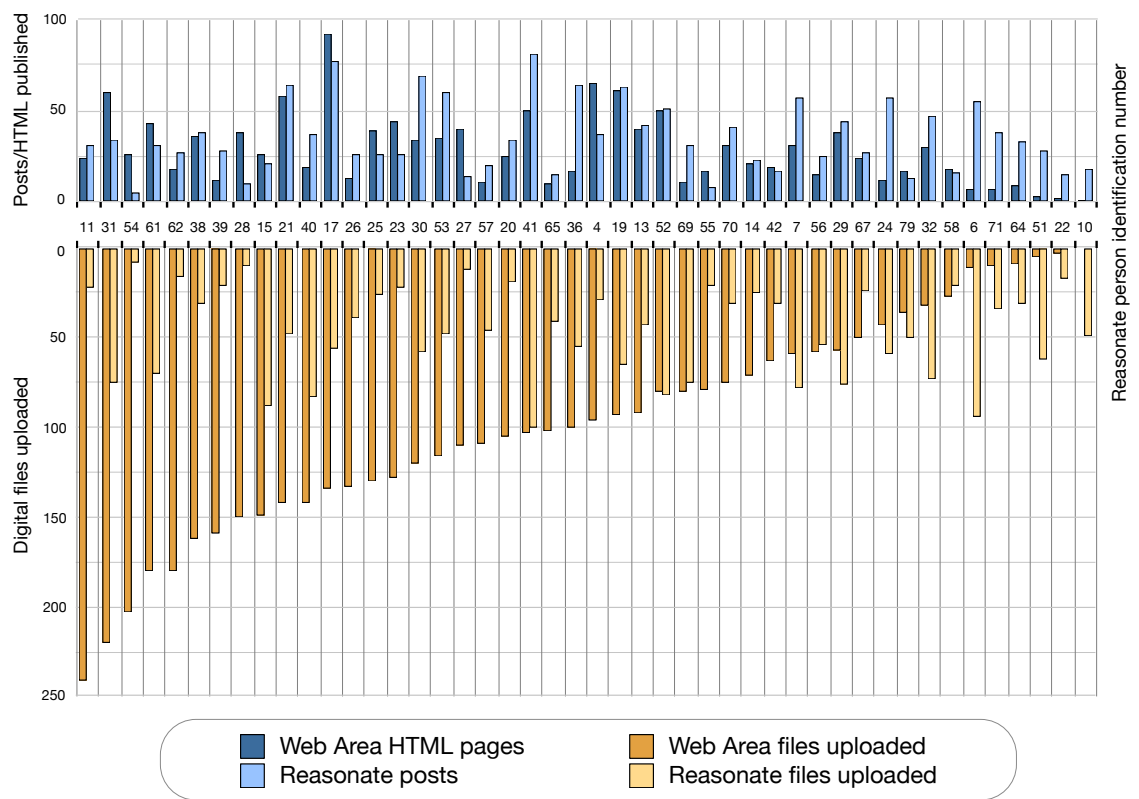


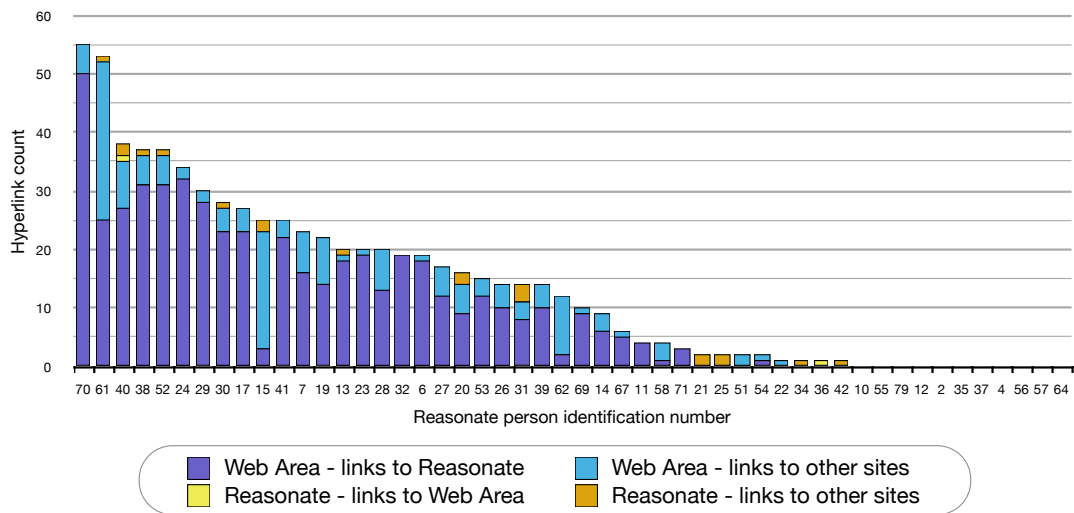
Figure 8.24: An ordered breakdown of the total amount of content published by each student to Reasonate's blog and Web Area sections



8.2.8.iii. The limited use of hyperlinks by most students

Only thirteen students embedded twenty or more hyperlinks into their Web Area websites that linked to blog posts in Reasonate or external websites (see Figure 8.25). This was less than expected considering most students created a relatively large number of blog posts and web pages.

Figure 8.25: An ordered breakdown of hyperlinks embedded into student content



8.2.8.iv. Summary

The demonstrated collaboration influence of decentralisation was strong. The majority of students who responded to the questionnaire accessed Reasonate from a location other than the School of Architecture. In addition, the students used the blogging and Web Area components of Reasonate to communicate very different messages. However, most students made limited, or no use of hyperlinks to weave these separate elements into a cohesive and coherent Building Story.

8.3. Discussing the influence of Hyperlinked Practice

The Reasonate software prototype tests demonstrated the majority of the principles' inferred collaboration effects. Although there were some limitations to these tests, the process provided new insights into how the functionality related to the principles could be more appropriately applied and tested in the future.

8.3.1. Reviewing the influence of the principles within Reasonate

Results from the testing of Reasonate indicated that the majority of the inferred collaboration effects of the Hyperlinked Practice principles were demonstrated. The following sections highlight important findings in these results, factors which contributed to them, shortcomings in the testing process, and functionality which could be incorporated into new digital tools to enhance the principles' influence.

8.3.1.i. *Situational awareness*

On average, the majority of students were contributing new blog posts at least twice a week, and 62% of respondents felt they often used Reasonate to review their project or the work of others. From the perspective of situational awareness this was a strong performance, as it indicated that most students were aware of the progress they and their fellow team members were making.

Reasonate's RSS and search functionality was intended to support the principle of situational awareness, but in practice this was not realised. The testing environment played an influential role in this outcome. For example, in the case of RSS there were three factors that contributed to its lack of use by students:

- **Software availability** - RSS news-readers that were available at the time of testing required the installation of third-party applications or web browser plugins. Due to a change in the configuration of the School of Architecture's computers, this was not possible. Consequently, if students wanted to use the RSS functionality they had to use their personally owned computers.
- **Low information quantity** - Even working in groups of three teams only produced on average six new blog posts per week. This generated little demand for the continuous monitoring capability provided by RSS.
- **The cohesiveness of the experience** - RSS allows content from a wide variety of information sources to be aggregated and consistently presented. The Reasonate interface automatically aggregated project information for students, which meant such functionality was not as necessary during the course.

Instead of utilising RSS, students most commonly alerted team members using the automatic email notification mechanism within Reasonate. Unfortunately, this behaviour

was only noted anecdotally, and not directly measured during the testing period, or investigated by the questionnaire.

The uptake of Reasonate's search functionality was much lower than expected, because there was limited time available to establish a sizeable search index, and once in place students had few opportunities to use it. As illustrated in Figure 8.3, the majority of students published less than 44 posts and 59 digital files, which was not a large body of work to recall, or manually browse with the help of the tagging and calendar-based navigation aids.

Although Reasonate's RSS and search functionality was unused by students, the course coordinator extensively used both functions. The coordinator monitored each team's RSS feed for progress, and identified those that were experiencing difficulties or not making progress. The coordinator used the search tool after the course was complete to locate relevant posts and digital files for review purposes. This usage pattern reflected the need for the coordinator to monitor and review a greater quantity of information from many different sources. Furthermore, third-party software could be installed onto the coordinator's computer to support these tasks.

These findings suggest that under light loads much of the functionality intended to support situational awareness will go unused if users can easily browse content. However, once the level of new information exceeds the individual's ability to remember or manually review all relevant collaboration information, the benefits of these functional aids are more likely to be realised.

8.3.1.ii. Ubiquity

The development of Reasonate as a web application made it possible for students to easily access and interact with the prototype from any Internet-connected computer irrespective of the software installed on the computer. Students took advantage of this capability, with 66% responding that they had interacted with Reasonate from their place of residence (see Figure 8.6). These numbers were supported by the analytical data, which showed that students regularly contributed content well into the evenings (see Figure 8.5). However, the analytical data collected during testing only measured what was contributed to Reasonate, and it did not take into account those that were only consuming content. This additional

information would have provided further analytical evidence to validate the questionnaire responses.

The second measurable influence of ubiquity was the decision by students to contribute the majority of their project information in digital formats that could be easily accessed by fellow students and interested parties. The small number of text-based documents (Microsoft Word and PDF files, see Table 8.3), was unsurprising considering Reasonate's blogging functionality satisfied the majority of these communication needs. Additionally, the blog format presented the information in a manner that was easier and faster to consume. There were a number of PowerPoint documents created but these were related to the interim hand-in for Assignment One, which stipulated this format.

Images were the most commonly used digital file medium because they were presented directly in the web browser. When reviewing Reasonate content this was a significant advantage because their accessibility provided an uninterrupted experience. In contrast, digital models could not be viewed within the browser, and opening them was often a time consuming process because it required a third-party application. The tendency to communicate using digital imagery was also supported by parts of the assignment work, which focused on the creation of rendered images by light simulation software. When teams were constructing their digital models, between April and early June, the growth of digital image contribution slowed, possibly because the CAD and BIM software tools did not have the same capacity to generate interesting digital images.

On average, each student uploaded approximately six digital models to Reasonate over the duration of testing. This number may have been greater if the majority of students had not been working on the School of Architecture computer network. Instead, most students bypassed Reasonate and used one of the University's shared network drives to exchange their large digital models. As a consequence, Reasonate tended to be a repository for completed digital models, rather than a mode of transferring work in progress between team members.

8.3.1.iii. Comprehension

The three weeks it took for students to reach acceptable levels of blog post and digital file contributions was considered a successful demonstration of the principle of comprehension. The assessment of the influence of comprehension within Reasonate could

have been enhanced by including a control group for the comparison of adoption patterns. Instead, adoption was measured by the time it took students to begin consistently contributing 16 or more blog posts per day, which represented the average number of blog posts published on each day of testing (1,818 posts in 112 days, or 16.23 posts/day). This daily average was a reasonable number, because it implied the typical student was publishing just over two posts per week (see Table 8.2). As described in Section 8.2.4.ii, the number of blog posts published after eighteen days of testing (18th March) was regularly surpassing 16 per day. This was a positive outcome considering the first week of testing was spent introducing the students to the course and its associated software. Coursework deadlines did not influence the adoption of Reasonate, because the first compulsory hand-in was not until the 28th March (see Figure 7.16).

Reasonate's tagging functionality struggled to gain adoption during testing. As illustrated in Figure 8.11, it was not until near the end of the course that approximately 20% of the students started applying tags in any reasonable quantity. In this seventeen day period between the 25th May and 10th June, 499 tags were created, which represented 50.4% of the overall total. Low adoption was not limited to students who had contributed relatively few blog posts or digital files. Of the ten project teams who published 50 or more blog posts, four used less than 17 tags to organise their content (see Figure 8.13). This indicates that the majority of the students did not understand or appreciate the functionality's underlying value. Two factors which contributed to this was the lack of an immediate need for tags, and the absence of a precedent on how they could be valuable in the future.

An organisational system such as tagging is only beneficial when the amount of potentially relevant information exceeds the ability of the individual to remember or manually locate it. Although the Reasonate testing process lasted long enough for students to generate a large amount of content, it was too short for the majority to forget what had been contributed, or for the quantity of information to become overwhelming. Locating content without the use of tags was helped by the ability for students to manually navigate content in Reasonate by person, team and the time it was contributed. In the majority of cases, this provided enough of a search matrix for students to locate relevant information. In addition, the nature of the assignment work meant that students were continually facing new design and technical challenges that bore no relationship to the problems previous faced. For example, the students' workflow progressed from constructing a digital model, to digitally rendering it, and finally presenting the outcome on a website. Consequently, the only time

students reflected on their entire body of work was when summarising their overall process for their personal website. Hence, it is likely to be no coincidence that tag use increased dramatically when most students began undertaking this task in the last week of May.

The lack of an established precedent on how students could derive value from tags was another factor that limited adoption. For many the notion of tagging was quite foreign, whereas Reasonate's blogging and digital file functionality was similar to the notebooks kept in other courses. As a consequence, many students did not understand, or see any value in, manually assigning tags when they could manually browse it. In contrast, the students who used the Blogger-based system a year later had a strong precedent of tagging, the content contributed to Reasonate in 2006. This work was still available on the Internet, and it is possible that some of those students who reviewed it in 2007 had a better understanding of the content they would produce, and a stronger appreciation of how tagging could be useful to them in the future. This knowledge may have influenced the adoption rate of tagging by the students in 2007, which as described in Section 8.2.6.i, was far higher. The adoption of tagging over time may also have been influenced by external factors, such as the increased presence of similar functionality in software commonly used by the students.

8.3.1.iv. Context sensitivity

Reasonate's project-centric functionality improved the effectiveness of many teams, and was considered a successful demonstration of the principle of context sensitivity. On joining a team, Reasonate's interface automatically changed so that the student could easily review published content that was specific to their project team, and be notified of any new contributions that were relevant to their project. Most students actively contributed to and reviewed their team's progress using Reasonate, with the majority of projects containing more than 37 contributed blog posts and 39 contributed digital files (see Figure 8.12). Emphasising content that was contextually relevant encouraged participation and review of the work, and 63% of respondents attributed the use of Reasonate to better teamwork (see Figure 8.2). In contrast to Reasonate, the Blogger-based system lacked project-related functionality, and fewer content contributions were recorded during its testing. This result could potentially suggest that an interface that is sensitive to the user's context promotes more interactions with it (see Figure 8.16).

8.3.1.v. *Emulative Modularity*

The comparable overall blog and digital file growth patterns between Reasonate and Blogger was a demonstration that a system that emulated another's core functionality could achieve and maintain a consistent collaboration dynamic. Although similar at a conceptual level, differences in the functionality and implementation of the two systems led to marked variations in some usage patterns.

The overall number of blog post and digital file contributions during the last twelve weeks of testing were similar for both systems. What separated the two systems was the very different adoption and utilisation rates in the first month. As illustrated in Figure 8.15, students adopted Reasonate relatively quickly, and 639 blog posts had been published by the 1st April. In comparison, over a similar period of time only 117 blog posts were contributed to the Blogger-based system. One anecdotal reason given for this difference was the inconsistency and complexity of the Blogger-based system compared to Reasonate. Whereas Reasonate had been designed and implemented specifically to meet the needs of the BBS303 course, the Blogger-based tool was created out of existing software products. Adopting this tool may have been more challenging because the experience was not as integrated, and its broader, more generic feature-set may have been more imposing to an inexperienced user. However, the data to further explore these issues was not available as the students who used Blogger in 2007 were not formally questioned on their prior Internet experience, or their adoption and usage patterns.

Although there were fewer blog posts published to the Blogger system, they were generally longer when compared to the contributions made to Reasonate. As illustrated in Figures 8.17 and 8.18, 82% of blog posts to Reasonate were less than 100 words in length, compared to 45% of Blogger contributions. One practical explanation for this was the different way the two systems handled digital file attachments. As described in Section 7.3.1.ii, digital files in Reasonate were treated as separate entities that were not part of the blog post's body of text. In contrast, Blogger was a traditional blogging tool that required references to digital files and thumbnail images to be embedded within the body of the post. The need for these references increased the word count of Blogger posts considerably, especially in cases where multiple digital files were uploaded. For the purposes of analysis these digital file references could not be removed, because they were deeply and inconsistently embedded into the Blogger content. Hence, many of the Blogger posts contained fewer words than the analysis indicates. Another factor in the different size

of blog posts may have been the setting of a precedent by the students who used Reasonate. Potentially, the students who took the course a year later may have had a better understanding of what was expected of them, and how their contributions to Blogger could support their coursework hand-ins, due to the previous work by the Reasonate students. However, no formal student surveys or analyses were undertaken at the end of the 2007 course, and so it was not possible to definitively identify the potential reasons for the different results.

The most striking difference between the two systems was the total number of tags created within the Blogger system compared to that of Reasonate. There is a possibility that Reasonate set a precedent that enabled students using Blogger to overcome the comprehension issues described in Section 8.3.1.iii. Another factor was the inability of Blogger to automatically group content by project. Instead, students had to manually tag their post so that it was identified as part of their project, which had two consequences:

- Approximately 300-500 tags were created in Blogger that were unnecessary in Reasonate because blog posts were automatically grouped by project.
- The need to tag content with their project identifier may have provided the impetus necessary for students to assign further tags.

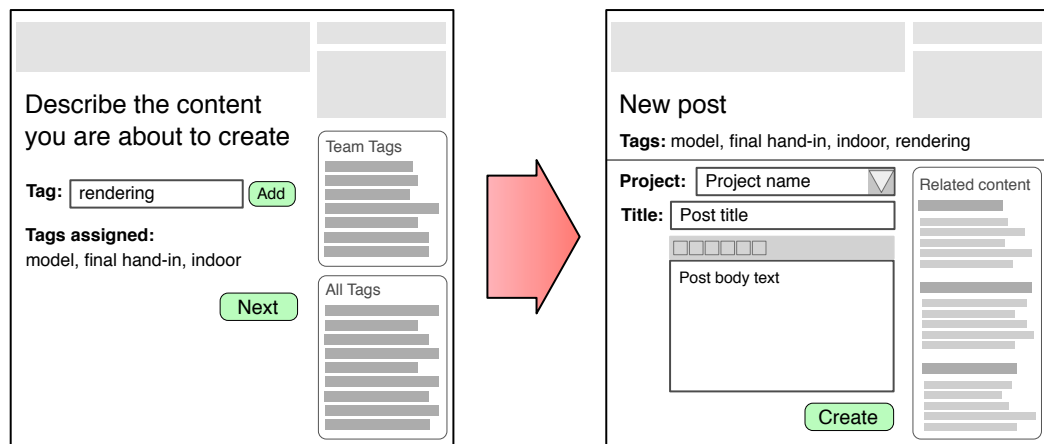
These factors could at least partly explain why 2.7 times as many tags were created by the students who used Blogger. Potentially, a more definitive explanation may have been obtained if the research could have been expanded to conduct a more detailed comparison of the tags created in both systems, and collect more in-depth information on the students' motivations and uses of tagging within Blogger. However, this was not feasible, due to the need to prioritise time and resources in this research.

8.3.1.vi. Emotive semantics

The collaboration influence of emotive semantics was difficult to demonstrate within Reasonate, due to issues related to the use of tagging within the architectural collaboration tool. As described in Section 8.3.1.iii, Reasonate's tagging functionality may have witnessed poor adoption because students had no immediate need to categorise content, and were unable to perceive how it could be useful in the long-term. This problem may have been addressed if the order of the content creation process had been reversed. As illustrated in Figure 8.26, the interface could be revised to encourage students to tag their

content before writing it, which could promote the application of tags, and enhance reflection by listing existing, relevant content alongside the unpublished blog post.

Figure 8.26: A potential Reasonate interface change to promote tagging



The application of tags increased during the last weeks of Reasonate testing (see Figure 8.11), but in total only nine students created more than 30 tags (see Figure 8.16). A similar result was recorded in Blogger a year later, when 43% of the tags were created by 10% of the users. In addition, there was almost no relationship between the number of blog posts published by Reasonate or Blogger users and the number of tags they created (see Figure 8.21). After examining the usage patterns of users of the social bookmarking service Delicious, Golder and Huberman (2006) came to a similar finding. They found “there is not a strong relationship between the number of bookmarks a user has created and the number of tags they used in those bookmarks... Some users have comparatively large sets of tags, and other users have comparatively small sets” (Golder & Huberman, 2006, p. 202). These three separate sets of results suggest that tagging is a polarising piece of functionality, that a subset of users will use extensively, and the majority lightly, or not at all.

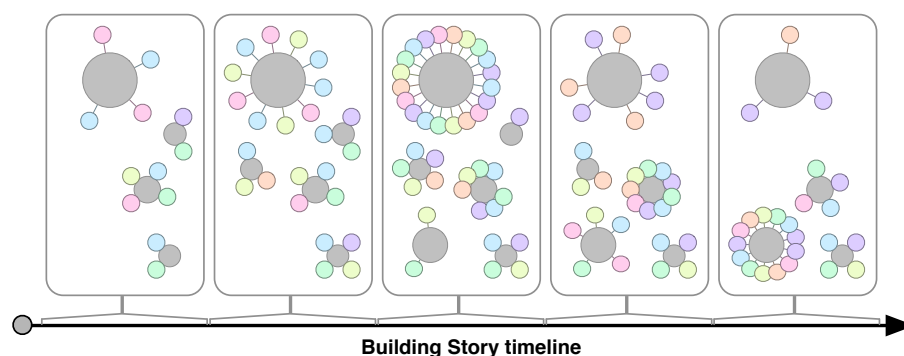
The overall semantic structure formed during the testing of Reasonate comprehensively described the course’s events, activities and issues (see Figure 8.22). However, for the typical Reasonate user this tag structure was difficult to understand because of the misspelling of terms, synonyms and team specific conventions (see Section 8.2.7.ii). This comprehension problem may have been a factor in the limited use of tags to organise and locate content. Reasonate’s user interface was modified several times during testing to address this problem (see Section 7.4.3), but overall these changes had little influence.

Based on these results and observations, three additional user interface changes may have improved the influence of emotive semantics:

The automatic grouping of tags by related terms - The tag structure illustrated in Figure 8.22 is easy to comprehend and navigate, but to be of value during architectural collaboration it must be automatically generated. Van Damme, Hepp and Siorpaes (2007) describe a method of constructing this ‘FolksOntology’ through the analysis of semantic relationships in social networks, lexicons and Semantic Web resources. Whilst a challenging undertaking, generating project-specific FolkOntologies could be possible with the aid of existing architectural lexicons (Woestenenk & , 2002) and analysis tools.

The time-sensitive presentation of tag lists - Each tag within the Building Story has four characteristics, its semantic meaning, the content it is related to, who created it, and when this action occurred. However, Reasonate’s user interface overlooked this last characteristic of time for the tagging functionality. This hindered situational awareness, because students were unable to understand how the project’s semantic structure was evolving in time. A more appropriate user interface would allow users to explore the evolution of the project’s FolksOntology, as illustrated in Figure 8.27. This interface would provide an excellent overview of the project’s Building Story, its issues, and a timeline of important events.

Figure 8.27: Visualising the evolution of a project’s semantic structure over time



Semantically related tag suggestions - Reasonate’s tag suggestion functionality used a simple, text-based comparison of user input to suggest potential tags. A more powerful approach would be to perform a semantic analysis of the input and content being tagged, so that more appropriate suggestions could be provided. These intelligent suggestions would potentially reduce the number of tag synonyms, and promote the use of terminology that was more appropriate for the project.

8.3.1.vii. Decentralisation

The ability for students to access Reasonate from any web browser, and the creation of two different information experiences, the blog and Web Area, were two different yet successful demonstrations of the principle of decentralisation. Reasonate's web-based interface allowed students to review and contribute content from locations outside of the School of Architecture. Whilst doing so was voluntary, 66% of students responded that they accessed Reasonate from their place of residence (see Figure 8.6). The prototype was not interacted with from other locations as often, but this was unsurprising considering most students were studying full-time at the School of Architecture, which is based 2km from Victoria University's main campus. Unfortunately, the web access logs from the Reasonate testing period were not stored or analysed, as these logs would have helped to validate the perceived access patterns.

Reasonate's separate blogging and Web Areas provided students with two related, but independent means of communicating. As illustrated in Figure 8.23, the students utilised these content areas in very different ways, and at an individual level there was little or no relationship between the quantity of content a student contributed to one area, compared to the other. This indicates that most students adopted very different communication strategies within the two areas. Overall, the average amount of content contributed to each location reflects these different strategies. In the case of the Reasonate blog, on average each student contributed 35 blog posts and 46 digital files, compared to the Web Area's average of 28 HTML files and 97 digital files. This analysis suggests that students used the Web Area primarily to communicate digital files, whereas a more balanced approach was used when blogging. Two technical issues must be taken into account when making these comparisons. Firstly, the total number of digital files contributed to the Web Area included website-related images, which varied in number depending on the student's design. Secondly, it was easier to upload large numbers of files to the Web Area because its WebDAV interface allowed files to be automatically uploaded from Dreamweaver. In contrast, when using Reasonate's blogging interface students had to manually upload digital files one at a time.

Only 40% of the students created more than 10 hyperlinks from their Web Area to Reasonate blog posts which described their decision making processes (see Figure 8.24). The limited number of hyperlinks used by most students was disappointing, because an opportunity was missed to weave these disparate resources into a cohesive, yet

decentralised story. Instead of forming links between their Web Area and Reasonate blog posts, many students replicated or recreated this information. Such actions may have been influenced by the previous year's website hand-ins. These had been created by students who had not documented their ongoing process using a tool similar to Reasonate, and as such they were designed and implemented as self contained entities.

8.3.2. The Reasonate testing process

Certain aspects of the Reasonate testing process limited the recorded influence of the principles and potentially biased aspects of the results in certain cases.

8.3.2.i. The implementation of the questionnaire

Integrating the questionnaire into Reasonate as a digital form was likely a factor in the relatively high response rate of 61.5%. Given the ease by which the questionnaire could be distributed and completed by students, it was unfortunate that this method of research was not used more frequently during testing. Formally questioning students at regular times would have provided more insight into their evolving usage and thought patterns, and benefited a number of areas:

Testing of the principles - Measuring the influence of the principles of comprehension and situational awareness would have been improved if student responses were available from different points during testing. For example, the student's comprehension of the system and their tendency to reflect on contributions may have evolved during the course. Understanding these changes, and the factors that led to them, could have yielded a better understanding of the principle's influence over time.

Prototype feedback - As described in Sections 8.2.2.ii and 8.2.7.i, the tagging, RSS and search functions of Reasonate were poorly adopted, or not used at all by most students. These functional and instructional failings may have been identified and addressed sooner in the testing process if students were formally questioned at regular intervals.

Refinement of the questionnaire - In the case of RSS there was a technical barrier that meant the functionality could not be adopted. If this had been identified as a problem earlier in the testing process, then subsequent questionnaires could have focused on the potential of the functionality and its future implementation, rather than confirming its already established lack of use.

Sample size and response rate - Having students complete questionnaires multiple times during the testing process would have likely led to an increase in the overall response rate, and may have been more likely to capture input from students who did not complete the course, or chose not to complete the questionnaire after the semester had come to an end.

8.3.2.ii. Reasonate's overall influence on collaboration

As described in Section 8.1.3, 63% of students felt that Reasonate improved their team's ability to work effectively. It is likely that this result was tempered by the close proximity of students to each other during the course. Most of the students were studying full-time at the School of Architecture, which was located within two adjoining, multi-story buildings. Consequently, the majority of collaboration interactions would have occurred in person, because there were very few physical barriers between students. As a result, Reasonate was generally used to record and review decisions which had already been made. If such close contact had not been possible, there is a high likelihood that student responses to collaboration-related questions would have been more polarised, because Reasonate would have played a more prominent role in enabling collaboration. However, whilst the nature of the testing environment was not ideal to test Reasonate's direct influence on collaboration, it did ensure that the prototype was focused on the recording and reviewing of Building Stories.

8.4. Summary

The Reasonate software prototype test successfully demonstrated the majority of the inferred collaboration influences of the principles of Hyperlinked Practice. The recorded influence of the principles of comprehension, ubiquity, emulative modularity and decentralisation was demonstrated to be particularly strong, with the majority of students rapidly and consistently adopting the blogging and digital file upload functionality, and seamlessly interacting with the different tools in Reasonate from multiple locations.

The influence of the principles of situational awareness and context sensitivity was less evident. Although it was demonstrated that students consistently contributed and reviewed project specific work on Reasonate, there was limited use of Reasonate's RSS search and tagging functionality to monitor and navigate relevant contributions. Due to the low adoption of Reasonate's tagging functionality, it was difficult to assess the influence of the principle of emotive semantics. However, the limited number of semantic structures that were formed were comprehensive and indicative of the content and issues experienced during the course.

One of the key strengths of the Reasonate prototype test was that it was designed and implemented specifically to meet the needs of the students in their course-work. This led to rapid adoption, high utilisation rates and a high response to the survey, which generated a sufficient amount of quality data for the analysis. There were some limitations to the testing process including institutional barriers to using the RSS functionality, the absence of a control group for comparing digital tool adoption rates and the absence of comparable survey information from the students who utilised the emulative Blogger system in the following year. Nevertheless, the process of testing Reasonate highlighted where functionality could be incorporated to enhance the principles influence, such as developing an interface that encouraged and automatically grouped tagged content in an easily navigable form to increase the influence of emotive semantics.

The next chapter explores the validity of the principles of Hyperlinked Practice further, by applying them within two thought experiments. These intent of these thought experiments was to investigate the usefulness of the principles in guiding the development of digital collaboration environments that facilitate Hyperlinked Practice.

9. Exploring the Potential of Hyperlinked Practice

Laying the foundation for digital knowledge bridges within architecture teams

The purpose of this chapter is to demonstrate that the principles of Hyperlinked Practice can be used to evaluate the architectural collaboration potential of emerging digital technologies. This evaluation process will identify the compatibility of an emerging technology with Hyperlinked Practice, and its potential collaboration influence should it be appropriately implemented and utilised within a project team. The Reasonate software prototype demonstrated that a collaboration tool embodying the principles enabled project teams to more effectively record and reflect upon the design process, but achieving industry-wide Hyperlinked Practice requires broad and long-term changes in collaboration processes. It is therefore important that the principles can act as an evaluation framework, which promotes the design and implementation of digital collaboration tools and strategies for facilitating Hyperlinked Practice.

To demonstrate the usefulness of the principles as digital collaboration evaluation aids, this chapter uses thought experiments to apply them to two emerging digital technologies, micro-blogging and social networking. These digital technologies were used as the subject of the thought experiments because at the time of the research they were relatively new communication concepts that were seeing widespread use by consumers, but had yet to be adopted within architectural project teams. This presented an opportunity to leverage the principles to better understand the digital collaboration potential and shortcomings of these emerging technologies. This principle-driven analysis process illustrated that appropriately designed and implemented micro-blogging and social networking tools could successfully contribute to the Hyperlinked Practice environment within a project team.

9.1. Achieving Hyperlinked Practice by applying its principles

The principles of Hyperlinked Practice were applied to the concepts of micro-blogging and social networking in order to identify how they could be successfully employed within project teams to record and reflect upon the design process. The concepts of micro-blogging and social networking were selected for these experiments because they are promising consumer technologies, that have yet to be broadly or systematically adopted by the AEC industry. In addition, these technologies have strong conceptual links to Hyperlinked Practice, because they are both Web-centric communication tools that record

the opinions and actions of a group of networked people. To understand how these technologies could be used to record and reflect upon the design process, each was analysed to assess how it embodied the principles of Hyperlinked Practice. This analysis outlined the architectural collaboration potential of micro-blogging and social networking, and identified the functionality that AEC-specific implementations would require to more appropriately record and reflect upon the design process.

9.1.1. Ranking how the principles are embodied within a collaboration tool

The principles of Hyperlinked Practice can guide the development of a digital collaboration environment where the design process can be comprehensively recorded and reflected upon. Evaluating a digital collaboration tool involves the assignment of ratings based on how thoroughly it embodies each principle. These ratings allow comparisons to be drawn between alternative systems, and they highlight aspects of the tool that need improvement if it is to facilitate Hyperlinked Practice. A five-step rating system was developed that could describe, in general terms, how thoroughly a principle is embodied within a collaboration tool (see Table 9.1). Due to considerable diversity in the design and implementation of digital tools, establishing how thoroughly the principles of Hyperlinked Practice are embodied within a tool is based on an informed opinion, rather than precise calculations.

Table 9.1: The rating system for how thoroughly the principles of Hyperlinked Practice are embodied within a digital tool

Situational Awareness	
Is the system capable of gathering and responding to external information generated by other systems within the project team?	
0 - Isolationist	The tool is isolated from the outside world and its processes and interface cannot respond to changes in this environment.
1	With significant effort the tool can monitor a few external resources so that its processes or interface can respond to changes in them.
2	With moderate effort the tool can monitor some external resources so that its processes or interface can respond to changes in them.
3	With minimal effort the tool can monitor a large number of external resources and can automatically respond to changes in them.
4 - Hive mind	The tool is deeply intertwined with its surrounding environment and its processes and interface automatically responds to changes in it.

Ubiquity	
Can the entire project team access the system from the digital tools that they commonly use?	
0 - Exclusive	The tool is only used by a single party and employs non-standard, proprietary technologies and data formats.
1	The tool has some industry use, but it is not readily available and employs non-standard, proprietary technologies and data formats.
2	The tool is readily available, but not widely used and generally employs non-standard, proprietary technologies and data formats.
3	The tool is readily available and widely used, but it generally employs non-standard, proprietary technologies and data formats.
4 - Universal	The tool is readily available, widely used and employs freely accessible technologies with standardised data formats.
Comprehension	
Is the system relatively easy to understand and use by those within the project team?	
0 - Enigma	The purpose, processes and outcomes of the collaboration tool are impossible to understand.
1	One or two aspects of the tool's purpose, processes and outcomes are somewhat understood by a few users.
2	After significant amount of effort, the tool's purpose, processes and outcomes can be understood by the minority of users.
3	After some effort, the purpose, processes and outcomes of the tool can be largely understood by the majority of users.
4 - Obvious	The purpose, processes and outcomes of the tool are readily understood by all users.
Context Sensitivity	
Does the system understand the hierarchy and ongoing activities within the project team, and can it tailor its operations and user-interfaces accordingly?	
0 - Oblivious	The tool has no understanding of the project situation and its processes and interface only operate one way.
1	The tool has no understanding of the project situation, but with significant effort, its processes and interface can be tuned.
2	The tool has a very limited understanding of the project situation, but with moderate effort, its processes and interface can be tuned.
3	The tool has a limited understanding of the project situation, and in response can change some processes and interface aspects.
4 - Aware	The tool has a strong understanding of the project situation, and in response automatically changes its processes and interface.

Emulative Modularity

Can the system's functionality be extended or replicated without interrupting the experience?

0 - Sculpture	The tool is made from a single, large component whose functionality cannot be extended or replicated.
1	The tool is made from a single, large component, but with significant effort minor functional aspects can be extended or replicated.
2	Parts of the tool are modular and with significant effort some its functionality can be extended or replicated.
3	The majority of the tool is modular and with some effort most of its functionality can be extended or replicated.
4 - Lego	The tool is completely modular and with minimal effort all of its functionality can be extended or replicated.

Emotive Semantics

Can categorisation within the system change over time, so that participants can record and navigate information in ways that relate to the project's current state?

0 - Meaningless	The tool employs no semantic system to organise the data it collects or transfers.
1	The tool employs a single semantic system that cannot be modified without considerable effort or planning.
2	The tool employs a single semantic system that can be modified with minimal effort or planning.
3	The tool employs multiple semantic systems specific to the user and their context, but modifying them requires considerable effort.
4 - Expressive	The tool employs multiple semantic systems specific to the user and their context, and if need be they can be easily modified.

Decentralisation

Can the collaboration interactions reliably occur without the presence of a central mediator?

0 - Monolith	The tool in its entirety is bound to a single location and cannot be moved or used anywhere else.
1	The tool is based in one location, but with significant effort it can be deployed to and used in multiple locations.
2	The tool relies on some centralised components, but with moderate effort it can be deployed to and used in multiple locations.
3	The tool has a few centralised components that do not stop it from easily being deployed to and used in multiple locations.
4 - Mesh	The tool's components are distributed and replicated, which presents no single point of failure and allows its use from anywhere.

9.1.2. Visualising how the principles are embodied within a collaboration tool

The principles of Hyperlinked Practice provide seven axes against which a digital collaboration tool's relative collaboration performance can be evaluated and compared. Visualising these results on a spider diagram is an effective means of illustrating a collaboration tool's overall capacity to support Hyperlinked Practice (see Figure 9.1). As an example of how this visualisation method can be applied, Figure 9.2 illustrates how six commonly used collaboration tools embody the principles of Hyperlinked Practice.

Figure 9.1: Visualising the ratings assigned to each Hyperlinked Practice principle

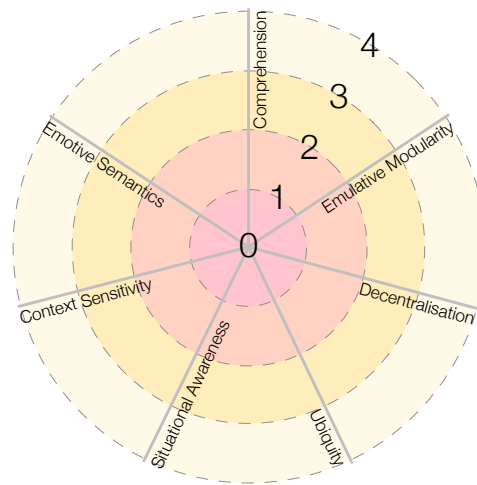
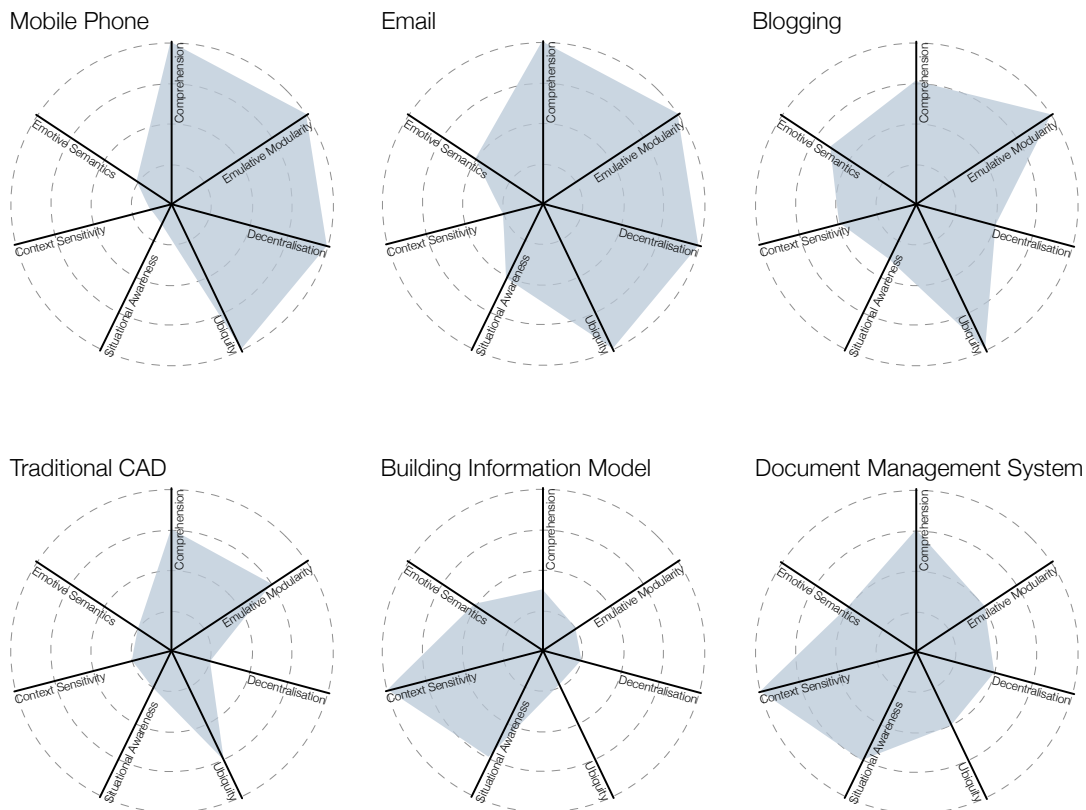


Figure 9.2: The embodiment of Hyperlinked Practice within current collaboration tools



In the spider diagrams of Figure 9.2, the following logic was applied to determine how each collaboration tool embodied the principles of Hyperlinked Practice:

Mobile Phones are readily available, well understood and can be used anywhere. They have no understanding of the project, and are used to conduct simple, verbal conversations.

Email is well understood and heavily used within the industry. However, email clients generally have little understanding of the project, or the activities going on within it.

Blogging is based on ubiquitous web technologies and is well understood, but it generally consists of standalone pages that are stored on a single server.

Traditional CAD is commonly used and understood within the industry, but it is generally considered a standalone tool for recording simple 2D and 3D data.

Building Information Models have rich semantic structures that enable the efficient manipulation of data, but few people can interact with a BIM due to its complexity.

Document Management Systems can monitor and respond to a variety of conditions, but they are generally centralised services that are unique to a project.

9.1.3. Applying the principles within digital collaboration thought experiments

When applying the principles of Hyperlinked Practice to micro-blogging and social networking, these five questions were addressed:

- **What** is the underlying technology and collaboration problem it relates to?
- **How** does this tool currently embody the principles of Hyperlinked Practice?
- **Where** could the tool be modified to improve its Hyperlinked Practice capabilities?
- **Why** would a version of the tool that embodies the principles benefit collaboration?

This line of investigation illustrated how the principles of Hyperlinked Practice could guide the development of AEC-specific implementations of each technology, and what effect these proposed tools would have on the recording of the design process.

9.2. Using micro-blogging to record architectural design conversation

The majority of professionals within the AEC industry use the telephone and email to collaborate on immediate design problems (Howard & Petersen, 2001). However, as described in Section 3.3.1, a disconnection exists between these conversations and the Building Information Model where the agreed upon architectural solution is recorded. As a consequence, it is difficult for a person interacting solely with the BIM to take part or learn from these conversations, because they are often oblivious to them taking place. Micro-blogging is an Internet-based communication medium that could provide the common thread to tie these disparate sources of project information together. It would achieve this through enabling the issues and outcomes discussed during architectural conversations to be quickly recorded by any member of the project team. In addition, those working on the BIM would be able to actively monitor and search across these conversations to gain a better understanding of the unfolding design process. Consumer micro-blogging services embody many of the principles of Hyperlinked Practice, but they are not tailored to meet the specific demands of architectural collaboration. An AEC-specific implementation of micro-blogging would address many of these shortcomings, and would be a highly capable means of recording the design process. However, for industry adoption to occur, AEC-specific micro-blogging systems must be integrated into the BIM toolset, so that participating in design conversations using this medium is part of the digital workspace.

9.2.1. Broadcasting and monitoring simple messages with micro-blogging

Micro-blogging is an emerging Internet-based communication medium that could significantly improve the timeliness and accessibility of architectural collaboration discussion. Popularised by the Twitter consumer web service, micro-blogging is the publishing of small messages that are broadcast to interested parties (“followers”). Due to the small and simple nature of the messages conveyed, users are able to read and publish messages using any network connected device. The technology has proven adept at conveying news and discussion amongst clusters of individuals who share common interests (Java, Song, Finin & Tseng, 2007, p. 61). Currently adoption is centred around consumer services, but efforts are underway to introduce the technology into businesses.

At a practical level, micro-blogging is the publishing of a short text message to an Internet service that then notifies interested parties and stores an archive of the message as a web page. The concise nature of these messages (~140 characters) allows them to be produced

and consumed by almost any device that is connected to a cellular network or the Internet. As a consequence, collaborators are free to participate at a time and place of their choosing. Users of a micro-blogging service “subscribe” to people who they are interested in receiving messages from, and “track” specific subjects that are published by the group. The risk of information overload is reduced by users explicitly defining their interests, as this increases their control over their incoming information flow.

9.2.2. Integrating the design conversation with the digital model

Micro-blogging could form a link between the ongoing design conversation and the project BIM. Teams that communicated using micro-blogging would be capable of recording the design process and its issues as the project BIM evolved. This would allow design issues and outcomes to be monitored, discussed, and understood by the entire project team.

Most collaboration tools assume the initiator of a conversation knows who should take part, and that those selected are able to participate at that time using the chosen medium. For example teleconferences are limited to those invited on the call at that time, and email conversations are received only by those who the messages are addressed to. In addition, these collaboration exchanges are self contained, and their outcomes require manual dissemination throughout the project team. In comparison, recipients of micro-blog messages are not explicitly defined, but rather they are inferred through a “follower” and search-based syndication process. This means that a recipient could be anyone who has expressed an interest in receiving the author’s contributions, or messages related to this subject. Due to the simple nature of these messages, recipients can receive and respond to these communications using any network connected device.

Beyond exposing internal conversations to the broader design team, an added architectural collaboration benefit of micro-blogging is that a web page is created for each published message. Each of these HTML documents have a unique address (URI), links to further information, the author’s details, and the date of publication. These documents become part of the project’s knowledge base, and can be browsed, referenced and indexed using existing web browsers or search engines. From the collaboration perspective this is important because it enables knowledge reuse, allowing new members to familiarise themselves with the project’s design history.

This scenario describes how an AEC-specific micro-blogging system could be used:

Anne was making some last minute design alterations to a project that was under construction. On selecting key elements within the project's Building Information Model, the software would automatically perform searches of the micro-blog archive for messages that linked to, or included tags relevant to, that aspect of the design. Results were presented to Anne in a window alongside the digital model. She had listed this information by significance, to ensure that messages from her direct superiors were prominently displayed. The majority of these messages dealt with historical aspects of the design process, but occasionally Anne came across micro-blogs published during the briefing process or onsite that highlighted issues she was unaware of. In this instance there were no obvious reasons why repositioning the wall would pose a problem, so she made the modification to the digital model and saved her changes. The software registered this as a significant change, and prompted her to record a micro-blog entry.

Anne entered: 'Repositioned the interior wall of office space C to satisfy the client's request.' To support her claim she included a link to the change request stored in the project's document repository. To alert relevant members of the team, the software automatically included a '#change-alert' tag within the message.

Anne's business partner Andy meanwhile was on his way to the project's weekly site meeting. His smartphone beeped with the arrival of Anne's SMS micro-blog message, letting him know that she had made the design changes and that the updated plans were available. Arriving onsite he found Frank the foreman had also received the message and had downloaded the updated plans to his laptop. Unfortunately, on inspection they soon realised that the recent and installation of a heating pipe would make implementing the proposed design change difficult. This pipe was modelled within BIM, but the installed pipe was significantly larger than what was represented digitally. In search of a compromise, Andy used his smartphone to photograph the problem area. He posted these images to his micro-blog, along with a few ideas Anne could explore. At this time Stan, the services engineer posted a micro-blog saying that clear access to the pipe was very important. Stan had moved on to a new job, but he had kept tracking the project for any messages about services, just in case a problem like this were to occur. Andy, Frank and Stan had a brief teleconference to discuss alternatives. After the conversation Frank used his laptop to post a few micro-blog messages that summarised the issues discussed. In the meantime Andy made his way back to the office. By the time he arrived he hoped that Anne would have digested all of this information and come up with a solution.

9.2.3. Micro-blogging within a project's information cloud

Micro-blogging is a more ubiquitous and easier to comprehend evolution of blogging, which was unsuited as a messaging service because of its complexity. However, whilst the principles of comprehension and ubiquity are deeply embodied within the technology, other aspects of Hyperlinked Practice are not as strongly represented.

9.2.3.i. Situational Awareness

Micro-blogging strongly embodies the principle of situational awareness because of its emphasis on presenting users with live data streams that are generated based on the list of people they are “following” and keywords they are “tracking” (Böhringer & Richter, 2009, p. 294). The message streams that micro-blogging systems generate are exposed in ubiquitous formats such as HTML, RSS and XMPP, which other applications can collect, present and act upon. Unfortunately the quantity of messages generated by a micro-blogging system means there is a significant risk of information overload. This situation is not helped by the immaturity of the applications that collect and display these messages, which means most lack features that could make monitoring these information streams more manageable. Third-party applications are also beginning to use micro-blogging services to automatically publish information based on external events. For example, it is becoming common for emergency service agencies to publish automated alerts and updates to Twitter, so that the local populations are made aware of the unfolding situation (Vieweg, Hughes, Starbird & Palen, 2010).

9.2.3.ii. Ubiquity

The simple conceptual and technical characteristics of micro-blogging enables its messages to be produced and consumed by almost any network connected digital device. This platform ubiquity ensures micro-blogging is accessible to the broadest possible audience. From a collaboration perspective this is important because it provides participants with the opportunity to passively monitor or actively take part in project discussion, irrespective of their location. At a technical level micro-blogging systems have leveraged ubiquitous communication protocols such as HTML, RSS and Extensible Messaging and Presence Protocol (XMPP) to output a user's messages. This has enabled the rapid growth the micro-blogging ecosystem. For example conventional search engines can crawl a micro-blog's HTML content, whilst newer ‘live’ search and trend services can monitor XMPP output in near real-time (Casey, 2010).

9.2.3.iii. *Comprehension*

Publishing 140 character long, plain text messages is an easy task to perform and technically implement. In addition, the premise that people “follow” other users or “track” ideas is well understood by a broad audience. This comprehension has led to very high adoption and usage rates of consumer micro-blogging services such as Twitter (Farhi, 2009). Whereas other technologies have increased their complexity in order to enable new functionality, the developers of micro-blogging systems have instead utilised easily comprehensible “hashtags” and hyperlinks within its plain text messages to add semantic meaning and references to multimedia content (Reinhardt, *et al.*, 2009, p. 147).

9.2.3.iv. *Context Sensitivity*

Contemporary micro-blogging servers and clients do not embody the principle of context sensitivity, and instead focus on publishing and presenting a contextless stream of messages. However, some messaging conventions have emerged that are allowing context to be reflected within micro-blogs. Two examples of this are the use of @ and # characters to identify people and topics (Starbird & Stamberger, 2010, p. 2). Many micro-blog clients and services are able to use this information to generate conversation threads and recommend relevant micro-blog content or users. Although this functionality is in its infancy, services like FriendFriend are using these techniques to enable “real-time conversation” (Kirkpatrick, 2009). Unlike a conventional conversations which involve a clearly defined group of people, these micro-blog based exchanges are the result of aggregating individual micro-blog messages into a single and easy to comprehend thread.

9.2.3.v. *Emulative Modularity*

Emulating the functionality of micro-blogging is a relatively simple task because the concept and its underlying technologies are ubiquitous and easy to comprehend. As a consequence, there is a growing number of micro-blogging implementations, all of which are capable of broadcasting 140 character long messages. Micro-blogging has also demonstrated its modularity through the rapid and diverse growth of the client software which can interact with the service. In addition, by using hyperlinks and semantic syntax (@ and #), users and software developers have added new layers of functionality to micro-blogging without breaking backwards compatibility. The first and most prevalent of these is the widespread use of URL shortening services that enable hyperlinks to easily fit within

micro-blogging's 140 character limit. In addition, many image sharing sites now allow images to be easily uploaded and displayed within standard micro-blog messages.

9.2.3.vi. *Emotive Semantics*

Micro-blogging does not explicitly define any semantic structures, or provide a technical means for users to create their own. However, a community-driven process has seen the growing acceptance of “hashtags” (#) as a means of assigning semantic meaning within messages (Passant, Hastrup, Bojars & Breslin, 2008, p. 3). Hashtags have been used to aid in the searching of micro-blog content, and to identify semantic trends within its communities. Although hashtags have provided micro-blogging with a flexible means of applying semantic meaning, including them within a message reduces the number of characters that can be used for content. Due to this trade-off, micro-blogs generally form shallow and broad semantic structures (Starbird & Stamberger, 2010, p. 3). This reduces the navigability of a micro-blog's semantic structure, because many of the higher-level semantic concepts that could link different terms are omitted in the interests of brevity. To address this shortcoming, initiatives are exploring how Semantic Web technologies such as Resource Description Framework (RDF) and Friend Of A Friend (FOAF) can be used within micro-blogging (Passant *et al.*, 2010). However, improving the ability of micro-blog messages to convey semantic meaning in this manner increases the technical complexity of the system.

9.2.3.vii. *Decentralisation*

Users can publish and receive micro-blog messages from any network connected location. However, the most popular micro-blogging services use a centralised service for broadcasting messages. In the case of Twitter, this centralisation has led to ongoing issues with reliability, due to the problems of scaling this broadcasting service to handle vast numbers of messages. Consequently, the move towards a more decentralised system, that enables different micro-blogging services to communicate with each other, is a paramount objective for “second generation” micro-blogging platforms.

“The model I am trying to follow is email. You have different servers that have different domains... But they are all interconnected, and as long as they are speaking the same simple protocol they work pretty well” (Prodromou, *et al.*, 2008).

These initiatives have led to the emergence of open standards for protocols, such as the OpenMicroBlogging specification (Passant *et al.*, 2010, p. 4). Although it is unlikely that consumer services such as Twitter will adopt these standards, their existence will support the gradual decentralisation of micro-blogging services.

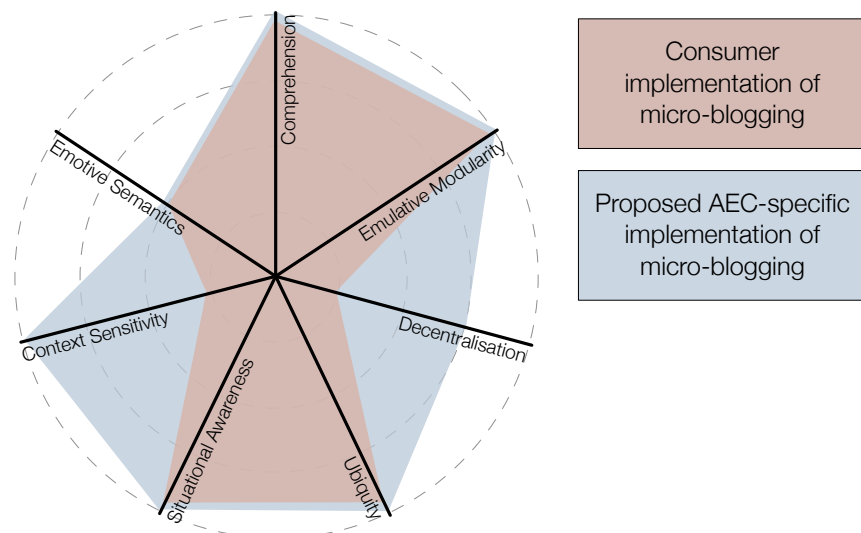
9.2.4. Developing an architectural collaboration micro-blogging platform

Current micro-blogging systems embody many of the principles of Hyperlinked Practice, but they generally display minimal understanding of the context of the information conveyed, possess rudimentary semantic mechanisms, and are hosted by a single, untrusted entity. Consequently an AEC project team, using a consumer micro-blogging service, would struggle to comprehensively and efficiently record and reflect upon the design process. An AEC-specific micro-blogging solution must be capable of the following:

- Integrating the act of publishing and consuming messages into existing workflows.
- Understanding and respecting the hierarchy of the project team.
- Storing sensitive project information in secure locations.
- Operating reliably within a distributed and digitally fragmented environment.

Figure 9.3 illustrates how a system incorporating these functional changes would embody the principles of Hyperlinked Practice relative to a consumer micro-blogging service. The key differences would be in the level of embodiment of the context sensitivity and decentralisation principles.

Figure 9.3: Comparing the embodiment of the principles of Hyperlinked Practice within a typical micro-blogging service and an AEC-specific micro-blogging service



9.2.4.i. Context sensitivity

A large portion of the architectural design process centres around the project's BIM and CAD digital models. Micro-blogging should seamlessly integrate with the software tools used to create and interact with these digital models, so that design discussions can be seamlessly reviewed and participated in. Along with adding tools to publish, browse and search micro-blog content, the BIM and CAD software should also prompt participants to contribute messages when they perform specific actions, such as save a final copy of a model, or modify key aspects of it. To support this process, these tools should also make it simple for users to include screen captures and links to download the digital model when composing a message. To promote context sensitivity, the BIM and CAD software should proactively search the project's micro-blog archive in order to locate messages that are relevant to the current design activity.

Unlike consumer micro-blogging services, a system geared towards AEC professionals should demonstrate context sensitivity by comprehending and respecting the hierarchical nature of project teams. Instead of expecting participants to manually create "follow" lists, templates for these relationships should be available that are based on the organisation of the project team. As managers create these templates they would also allow assign responsibilities and security levels to team members. Maintaining this hierarchy would allow the people and topics "followed" by a participant to be automatically updated as the composition of the project team changed. This would save time, reduce the workload, and ensure the participants were exposed to the most relevant sources of information.

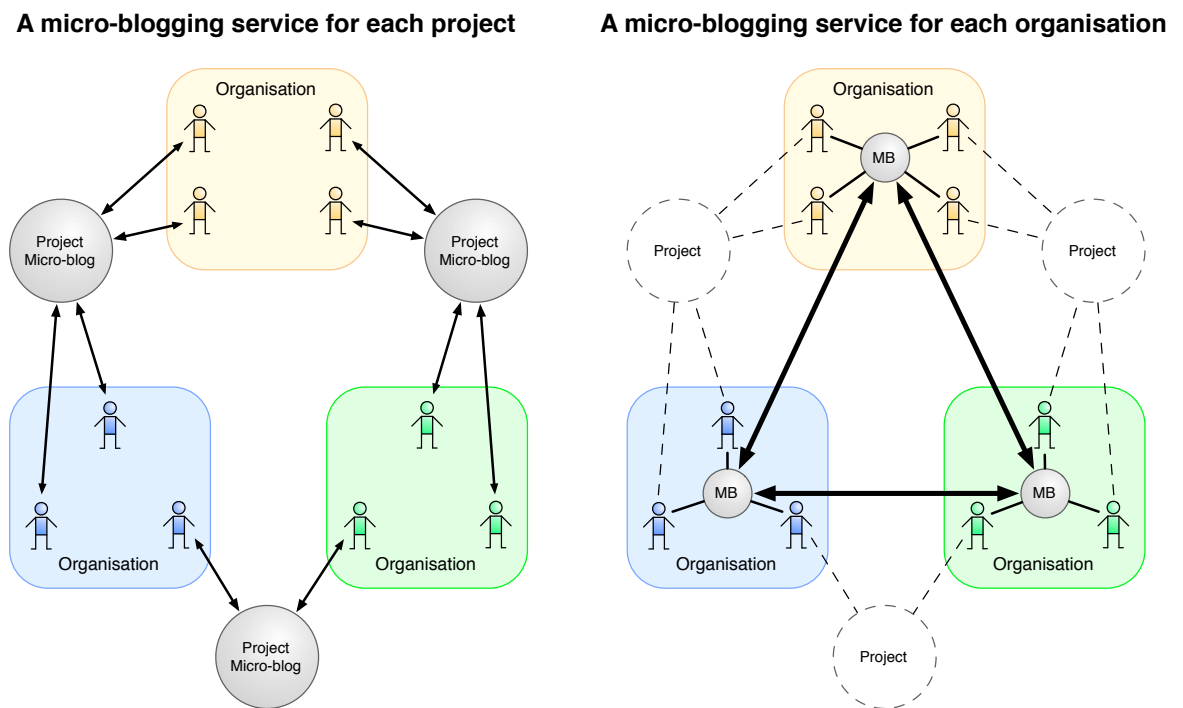
The AEC industry is a litigious environment, and any micro-blogging solution used within it must be capable of restricting access to published content if the context requires this action. Current micro-blogging services have limited security models (Böhringer & Richter, 2009, p. 296). For example, whilst it is possible to mark a message stream as private, if another user is granted access they can read every piece of content that is published by this account. Within an architectural project team, a finer grained security system is required, to filter access based on the context and content of the message. For example, an external consultant that joins a micro-blog conversation should only be able to view messages posted by team members that relate to a specific project. In addition, project administrators may wish to filter access based on specific topics or periods of time. For example, the consultant's access may need to be limited to messages published between a defined period of time about specific aspects of the design. At a technical level,

this context-level security would be applied within the micro-blogging servers, so that the client software could operate unchanged.

9.2.4.ii. Decentralisation

Architecture projects are often temporary collaborations between multiple organisations. A successful, AEC-specific micro-blogging implementation should allow participants to seamlessly collaborate, irrespective of the micro-blogging service and tools they may be using. Two strategies for decentralising micro-blogging are illustrated in Figure 9.4.

Figure 9.4: Increasing the decentralisation of micro-blogging through the use of project-specific and organisation-specific micro-blogging services



The first technically straightforward strategy would be to setup a micro-blogging service for each project (Figure 9.4). User access to such a system could easily be controlled, and it could be assumed that any messages published to that service were directly related to the project. The shortcomings of this approach is that the service would still be centralised, to an extent, around a single point, and a single party would essentially ‘own’ the service and any information that passed through it. In addition, if someone was involved in more than one project they would need to maintain a collection of different, project-specific micro-blogging accounts. The second strategy for decentralising micro-blogging would be to enable micro-blogging services based within different organisations to communicate with each other. As discussed in Section 9.3.2.vii, this decentralisation strategy is a primary goal

of second-generation micro-blogging platforms. The benefit of this approach is that each person would have a single micro-blogging account that was related to their organisation. In a similar manner to email, the responsibility for directing messages to relevant people within the project team would then fall onto each organisation's micro-blogging server. However, in an AEC-specific system this form of decentralisation would be made more challenging by the need to consistently apply the project hierarchy and context-level security settings to the different micro-blogging servers. Collaboration could occur without this data transfer, but the presence of this information would benefit the user experience, and improve the overall embodiment of the principles of Hyperlinked Practice.

9.2.5. Summarising architectural micro-blogging

AEC-specific micro-blogging stands to be a powerful and valuable architectural collaboration tool. By implementing the set of functional improvements that were identified by applying the principles of Hyperlinked Practice, an AEC-specific micro-blogging system would be a highly capable means of recording the design process. Implementing these changes would not be a straightforward task, but the underlying functional groundwork already exists within the broader micro-blogging ecosystem.

9.3. Bluestreak and the birth of a collaboration kernel

Reflecting upon the digital design process is difficult considering the number of collaboration interactions that occur within the team, and the diverse range of digital tools utilised. A collaboration kernel that weaves together these disparate interactions and tools into a more cohesive collaboration environment would allow design discussion, issues and decisions to be efficiently and reliably exchanged between team members and the digital tools they currently use. Consequently, this thought experiment describes how Project Bluestreak, a prototype social networking service from Autodesk Labs, could be transformed into an effective collaboration kernel. To guide this transformation, the principles of Hyperlinked Practice have been used to evaluate the existing service and identify areas for future development.

9.3.1. Seamless collaboration within a fragmented digital environment

A successful AEC digital collaboration environment brings multiple parties together so that they can productively work towards a satisfactory and achievable design outcome. During this process participants must engage in a variety of interactions between team members and the digital models used to describe the design. These interactions form an extended social network that encompasses the members of the team, and the digital model which describes the current design state of the project. These interactions, and the technologies commonly used to enable them, are summarised in Figure 9.5 and Table 9.2.

Figure 9.5: The digital collaboration interactions within an architectural project team

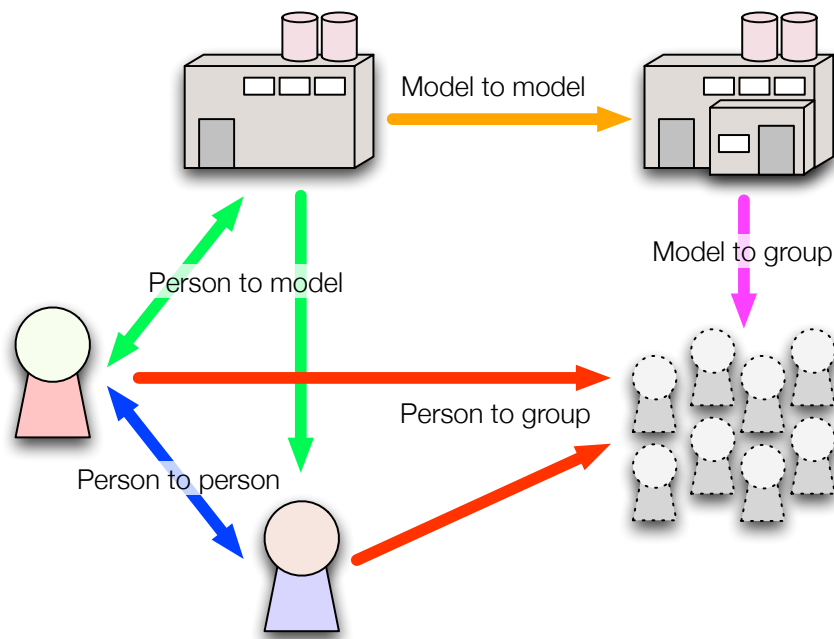


Table 9.2: The purpose, nature and enabling technologies of the digital interactions within an architectural project team

Person to person	
Purpose	Productive conversations between design participants are critical for the success of any design project. The intention of these interactions can be to present, question and debate all aspects of the design.
Nature	Given the non-linear and bi-directional nature of conversation, the ideas and data communicated are generally fluid and unstructured. To be most effective, the digital tools used should not introduce latency, as this can result in a disjointed conversation. During these exchanges it should be possible for participants to easily reference media such as images, documents, diagrams and digital models.
Enabling technologies	The most common person to person interactions during a design project tend to be physical meetings and telephone conversations. Internet-based voice and video conferencing technologies are also beginning to be adopted (Heidrich, 2007). In situations where person to person interactions are limited in scope, or do not warrant the interruption of a real-time meeting, email is commonly used to exchange thoughts and information.
Person to group	
Purpose	Individuals must be capable of efficiently and reliably communicating information about the design to the project team, such as its status, issues and design outcomes.
Nature	This interaction is uni-directional because a group cannot directly add to a conversation. If a recipient of a person to group message responds this spawns a new person to person, or person to group interaction. Person to group interactions typically have a specific topic, but the supporting media referenced during the exchange varies depending on the subject and its context.
Enabling technologies	Email is the most prevalent digital means of communication between a participant and the project team (Vollmer, 2005). Messaging systems and discussion forums embedded within project extranets, company intranets and the public Internet are used, but compared to email their industry adoption is limited (Becerik, 2004). Many document management systems include support for person to group interactions, but this is typically a secondary aspect of its functionality.

Person to model	
Purpose	A participant interacts with the model to understand the design, express new ideas and review the contributed work of others. If the participant cannot efficiently comprehend or manipulate the model, their ability to take part in the broader design discussion is impacted.
Nature	The nature of this interaction depends on the role and technical ability of the individual. It is common for the majority of an AEC project team to be unable to make changes to a model, either for technical or process reasons.
Enabling technologies	The primary interface between the individual and a digital model is the CAD and BIM software used to create it. Given the complexity and cost of these tools, it is common for the majority of the team to use freely available, DWF and PDF reader applications. In these cases, recipients cannot modify the digital model, but they can provide feedback in the form of text and graphical annotations.
Model to model	
Purpose	To simplify and distribute the overall process, a design is typically developed using more than one digital model. These distinct models must be efficiently and consistently integrated, so that the team can comprehend the overall design.
Nature	Given the technical complexity of this task, the flow of data in a model to model interaction typically is in one direction. This usually involves extracting the data in one digital model and merging it into a primary ‘master’ model.
Enabling technologies	Technologies for model to model interaction vary in complexity, capability and industry penetration. The most common means of consolidation is the manual importing of data from standard digital model formats such as IFC or DWG. Unfortunately, incompatibilities between different CAD implementations mean such interactions can lead to inconsistent import results.
Model to group	
Purpose	The overall design needs to be distributed amongst the project team for review and construction purposes. The information conveyed by the model is raw data that relates to the current state of the design, rather than personal opinion.
Nature	Given the physical and technical distribution of a project team, it is usually impractical for a group to interact with a digital model in real-time. Instead, snapshots of the model’s design state are generated and communicated in a format all interested parties can access. If group members wish to respond to the information conveyed, they must initiate a new person to person, person to group, or person to model interaction.
Enabling technologies	In larger projects, document management systems are commonly used to ensure the project team is informed of changes to the digital model and supporting documentation. Many of these services are integrated into the CAD and BIM tools, so that interactions between the model and group are seamless. In smaller projects where the cost and complexity of these systems cannot be justified, distribution commonly occurs via email or Internet-based file transfer.

Given these diverse functional requirements, no single technology is currently capable of satisfying the digital collaboration needs of a project team. This poses a problem because participants stand the greatest chance of receiving timely and relevant design information when the digital experience is well integrated. Unfortunately the boundaries between two or more collaboration tools generate inefficiencies, confusion and data loss due to the inability of many digital tools to integrate with each other (Froese *et al.*, 1998). This can often lead to the following issues:

Lack of Process Integration - The decisions or actions taken in one tool are often not reflected in others. In an ideal world, design decisions made during an email exchange would automatically generate outstanding to-do items within the digital model, and have the document management service (DMS) notify the team of forthcoming design revisions. When interacting with the digital model or DMS later in the project, this same trail of actions would be useful for understanding the motivations and justification behind design decisions. At present, these actions cannot be automatically undertaken, because there is no simple means of passing messages between the different collaboration tools in use within the project team is not available.

No Identity Management - Architectural collaboration tools typically do not use a single system for identifying users, or recording information about them. This forces participants to create numerous virtual identities and maintain a record of those used by other members of the team. This becomes problematic when reviewing a series of design decisions that have been made using a series of collaboration tools. For example, a project team that employs email, CAD software, and a Document Management System to collaborate on an architectural design will generally have to contend with three sets of identity information, as illustrated in Table 9.3.

Table 9.3: The identity systems used during a typical digital collaboration interaction

Interaction	Software	Identity System
Person to model	CAD software	An account on the local operating system. e.g. <i>COMPANY\username</i>
Person to person	Email	A globally unique email address. e.g. <i>participant.name@company.com</i>
Model to team	Document Management Service (DMS)	DMS-specific user account. e.g. <i>participant_name</i>

With three different identity systems, tracking a design decision from conception in email to its final distribution by the Document Management System is a complex process. Questioning a design decision can be difficult, as the participant must first identify who to contact and then discern which of the person's virtual identities is related to the collaboration tool.

Functional and Data Repetition - The lack of messaging or identity integration between collaboration tools results in the repetition of functionality and data-entry tasks. Common information such as the identities of team members, their project roles, and general interests cannot be easily shared or consumed by applications. Similarly, common collaboration functionality that is used by multiple applications must be continually reimplemented rather than being reused.

9.3.2. Using a collaboration kernel to integrate collaboration interactions

In an ideal world, the various collaboration interactions which occur during a project would be supported by a single, tightly integrated software application. This 'digital collaboration swiss army knife' would promote an efficient and cohesive collaboration environment by reliably recording and seamlessly communicating relevant design information throughout the team. Unfortunately a universal AEC digital collaboration tool is impractical, both now and in the foreseeable future, due to the complications which arise from bundling so much functionality within a tool intended for use by a diverse audience.

The most efficient and reliable means of addressing this problem is to establish a collaboration kernel that can act as an intermediary between the disparate architectural collaboration tools. This Internet-centric service would become the project's digital post office, overseeing the exchange of messages that support, summarise and promote the collaboration interactions taking place within the team. A collaboration kernel's presence would be subtle, but its influence on collaboration would be significant, as described in the following hypothetical scenario.

Pam the project manager reviewed the client's email. The design of the entrance foyer for their commercial development needed to be enlarged to accommodate more activities than originally planned. This was not a simple task because the layout of the ground floor was tight and allocating more floor area would require sacrifices. She selected the client's email, pressed the New Task button in her email client, and from the list of people in the project team she assigned the problem to Andy the architect. She wrote a quick summary of the task ahead:

***"From Pam to Andy:** Tomorrow can you identify an alternative foyer design based on the criteria listed in this email?"*

After pressing the 'Create Task' button, the application uploaded a copy of the email to the architecture practice's internal server where Andy could access it. It then sent Pam's message and a link to the relevant email to the collaboration kernel, which would ensure the task was brought to Andy's attention.

The next morning Andy arrived in the office and logged into the Practice's Intranet. His personalised homepage checked in with the collaboration kernel, which promptly returned the task Pam had assigned to him. Andy read the message and followed the link to the referenced email. Being newly assigned to the project, he was not fully aware of previous design decisions associated with the foyer. To provide some background he queried the collaboration kernel for all the design interactions related to that specific part of the building. The service returned a chronological history showing who had been involved in its design, and what input they had had. The breakdown revealed two particularly active design periods, which included references to early 3D models and preliminary spacial renderings. Reviewing this work and its associated discussions, Andy quickly came to terms with the design concepts and issues within this part of the building. He opened the project's BIM, and before starting work made the following note in his work-log:

***"From Andy to everyone:** I am spending this morning redeveloping the entrance foyer as per Pam and the client's instructions."*

He attached Pam's task to this note and saved it to the work-log. In the background, the BIM software published the message to the collaboration kernel. The kernel broadcast the message to everyone in the team to warn them that changes may be occurring within this aspect of the design.

Meanwhile, Lenny the lighting consultant was finalising the ground floor lighting design. He had received a phone call from the client requesting a change to some of the fittings, but the proposed foyer redesign had not been mentioned.

Lenny's lighting simulation software displayed a notification:

“From Andy: *I am spending this morning redeveloping the entrance foyer as per Pam and the client's instructions.”*

Lenny was unable to access Pam's emailed instructions as he worked in a different organisation, but he got the feeling this could influence his lighting design. He telephoned Andy, and very quickly they identified the change would pose a problem. After the call Lenny made some notes on the conversation, and the changes they had agreed to make within their respective digital models:

“From Lenny to everyone: *Andy and I have discussed the foyer changes, and have come to an agreement that will suit the client's and code requirements.”*

“From Lenny to Andy: *If you redesign the east side of the foyer as discussed I will be in a position to make the relevant lighting design changes this afternoon.”*

These notes were published to the collaboration kernel where they were distributed to everyone in the team. The second note was addressed to Andy in order to establish a record of their design agreement.

Andy spent the morning modifying foyer within the digital model. On completion he published the revised model to the project's document management system (DMS) for review. On committing the change he summarised the changes:

“From Andy: *This revision to the foyer takes into account the changes in capacity requested by the client. Accommodating this extra space required modifications to the surrounding design, which is forcing Lenny to redesign aspects of the lighting.”*

News of this change was automatically published to the collaboration kernel by the DMS so that members of the team who were tracking this model could be notified. Lenny received this update and downloaded the revised model for review. After confirming there were no conflicts with his lighting design model and that code requirements were met, he sent the following message:

“From Lenny to Andy and Pam: *I have reviewed Andy's proposed foyer changes alongside my revised lighting layout. Everything checks out fine.”*

The collaboration kernel delivered the message to Pam's mobile phone because she was at a site meeting. On receipt of the news she sent an SMS message in reply:

“From Pam to Andy and Lenny: *Good progress. When I get back to the office I will have the client review both changes.”*

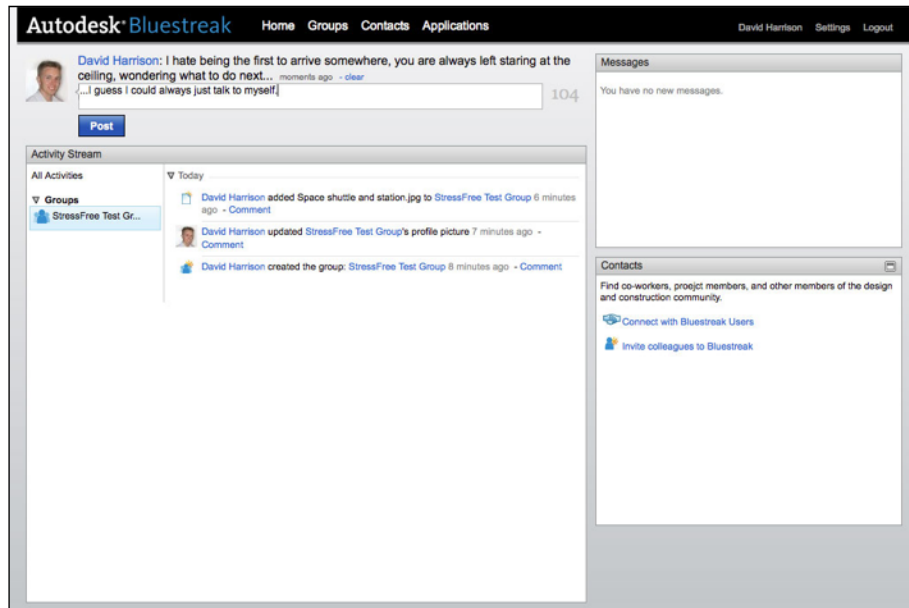
The SMS went to a service that automatically forwarded incoming messages from approved numbers to the collaboration kernel for distribution amongst the team.

Establishing a collaboration kernel and attaining this level of integration between the various digital tools used by the project team will take a significant amount of time and resources. To explore how the concept of a collaboration kernel could become a reality, the following thought experiment describes how Bluestreak, a social networking prototype released by Autodesk Labs in 2009 (Sheppard, 2009) could be transformed into one. The purpose of this exercise is to demonstrate how the principles of Hyperlinked Practice could be used to guide the redevelopment of an existing AEC software product, so that it is more capable of recording and reflecting upon the architectural design process. This thought experiment was conducted soon after Bluestreak's initial release, and was published on the Web for peer review (Harrison, 2010). The observations made in this thought experiment relate to Bluestreak as it was implemented at the time.

9.3.3. The untapped potential of Bluestreak

Autodesk Labs' Project Bluestreak is a web-based tool for exploring the applicability and usefulness of social networking technologies within design collaboration. An example of the Bluestreak user interface is illustrated in Figure 9.6. Whilst unique for Autodesk, this is not the first time social networking concepts have been applied within the AEC industry. For example, Vuuch and Kalexo are two established and functionally richer products (Wong, 2009). However, Autodesk is a dominant and pervasive presence throughout the world of digital design, and if testing proves successful, Bluestreak could permeate throughout their entire software portfolio. Including social networking functionality within their entire product line could significantly benefit the workflow of Autodesk's customers, and ultimately influence the direction of collaboration within the industry. In the shorter-term, a key differentiator between Bluestreak and its contemporaries is the support pledged to third-party application development on the platform. Developer ecosystems that leverage the information and relationships stored within social networks have achieved significant business traction (Shih, 2009). For example, Salesforce's AppExchange and Facebook's Application Directory are prominent examples of this strategy. In both cases, large numbers of independently developed applications have flourished due to the popularity of the underlying core service. A collaboration-centric application ecosystem would not garner the same levels of developer or media attention as their consumer-focused peers, but within the context of the AEC industry it would be a powerful platform.

Figure 9.6: The Project Bluestreak social networking web interface

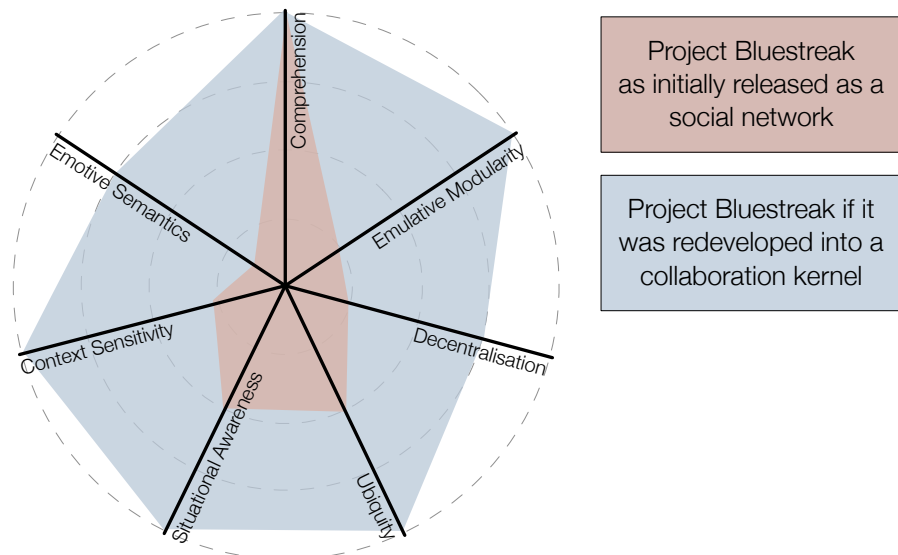


When viewed alongside the concept of a collaboration kernel and the principles of Hyperlinked Practice, Bluestreak as initially developed was a missed opportunity. Instead of a standalone website, the service should have been positioned as a social messaging service that was integrated throughout Autodesk's software portfolio. This would have exposed the service to a broader audience, and helped to position it as a potential collaboration kernel. Internally this would benefit Autodesk, because their various development groups would be able to leverage Bluestreak's collaboration functionality within their own applications via an Application Programming Interface (API). Once standardised, this API could be publicly exposed to enable integration with third-party applications, or entirely new collaboration experiences. It is likely that software developers would be eager to build on this platform, as it would simplify development and provide a direct, sanctioned link to Autodesk's applications and customer network. Transitioning Bluestreak from a standalone social networking tool to an integrated collaboration kernel would require a significant amount of redevelopment and new functionality. Rather than blindly working towards this goal, a more productive approach is to analyse how Bluestreak currently embodies the principles of Hyperlinked Practice. This analysis process will help to identify a set of functional improvements that are required if Bluestreak is to meet the demands of operating as a collaboration kernel and promote more effective collaboration

9.3.4. A Bluestreak in the project's information cloud

The intention of a collaboration kernel is to improve the timeliness and relevancy of information delivered to project participants. To achieve this, the kernel must provide a set of common functionality that can be easily leveraged by other AEC software tools. If Project Bluestreak is to fill the role of a collaboration kernel it should be developed to embody the principles of Hyperlinked Practice, because this would enhance its ability to be used by the entire project team to record and communicate the design process. An indication of how Bluestreak currently and potentially could embody these principles is illustrated in Figure 9.7.

Figure 9.7: Comparing how the principles of Hyperlinked Practice are embodied within the current and potential future implementations of Project Bluestreak



9.3.4.i. Situational awareness

As initially implemented Bluestreak had limited situational awareness, because it was dependent on manual data input and there was no way of externally monitoring the discussion taking place within the service. This was a considerable shortcoming because collaboration takes place over multiple communication channels, for example email correspondence and digital files that are uploaded to document management systems. A successful collaboration kernel should make the team aware of the collaboration interactions occurring within these external channels. The Bluestreak API could significantly improve situational awareness, as Autodesk and third-parties could create components that polled external services for state changes, or enabled data to be pushed

into Bluestreak from other applications. Examples of Bluestreak collaboration kernel components that could improve situational awareness are:

Changes - An agent that monitors files in a third-party document management service and informs the team when modifications take place. Most project documentation will not reside within Bluestreak, so knowing it has changed and to what degree is an important consideration during collaboration.

Progress - An agent that parses the project manager's Microsoft Project file and shared calendars so that it can alert the team of important events. The project timeline is continually evolving, and those involved cannot be expected to maintain it in multiple locations. Monitoring a project's timeline would also enable applications that are integrated with Bluestreak to be more contextually aware.

External Activity - An agent monitors an external email account, collaboration tool, or web service for information contributed by a third-party. A sub-contractor may not warrant full Bluestreak project membership, but they could be provided with an email address for submitting information and questions. The component could automatically monitor this account, and publish email received by it onto Bluestreak.

A collaboration kernel should also expose the data it collects to trusted third-parties. As initially implemented, Bluestreak users can manually monitor conversations via the website, or elect to have all status and group messages emailed to them. A more effective means of promoting situational awareness would be to expose this information in ubiquitous, machine readable formats such as RSS, XML and JSON. This would enable the collaboration interactions recorded in Bluestreak to be automatically processed and acted upon by software applications commonly used within the team.

9.3.4.ii. Ubiquity

Bluestreak has been developed using ubiquitous web technologies, which enables it to be accessed from any modern web browser. In addition, it places minimal restrictions on what information can be uploaded to the service or exchanged between participants. However, as implemented, there was no means of posting a message to the Bluestreak service without using the website's user interface on the website. A more ubiquitous approach would be to implement a micro-blogging API such as the one used by Twitter. This way

Bluestreak messages could be published and read using micro-blogging clients based on desktop computers and mobile devices (Makice, 2009).

Beyond promoting ubiquitous formats and processes, the concept of Bluestreak would need to become ubiquitous across Autodesk's software line, and portrayed as a collaboration umbrella that touches all aspects of Autodesk's activities. Conversations currently taking part within the Bluestreak web application need to be brought to the 3D CAD and BIM tools where the majority of design development, analysis and documentation is taking place. For example, when using Revit an architect should be able to review and participate in Bluestreak discussions without leaving the application. When the model is exported to DWF for sending to the contractor, relevant aspects of that discussion could be embedded into the file in order to preserve its design context.

9.3.4.iii. Comprehension

As initially implemented, Bluestreak was easy to understand as it had a minimal feature-set, and most of it emulated concepts made popular in social networking services such as Facebook. When making the transition to a collaboration kernel, the functionality should be kept concise, so that those using it have a clear understanding of what services it provides and why. Limiting Bluestreak's functional scope will help the platform gain adoption, because developers and end-users will be able to comprehend and appreciate its role in collaboration. This strategy has proven very successful for Twitter, which has flourished thanks to the ease by which developers and users alike have been able to understand its functionality, and how it can be leveraged to achieve the desired outcome (Chapter 10 Less Software: 37Signals, 2010).

Operating as a successful collaboration kernel requires that Bluestreak integrate with a diverse range of AEC tools. This requires a modular technical architecture that external applications and services can hook in to. A proposal for this technical architecture is illustrated in Figure 9.8, and integration with it can occur in three ways:

Components - Autodesk and third-parties will build components on top of the Bluestreak API that will form a critical part of its web interface and functionality.

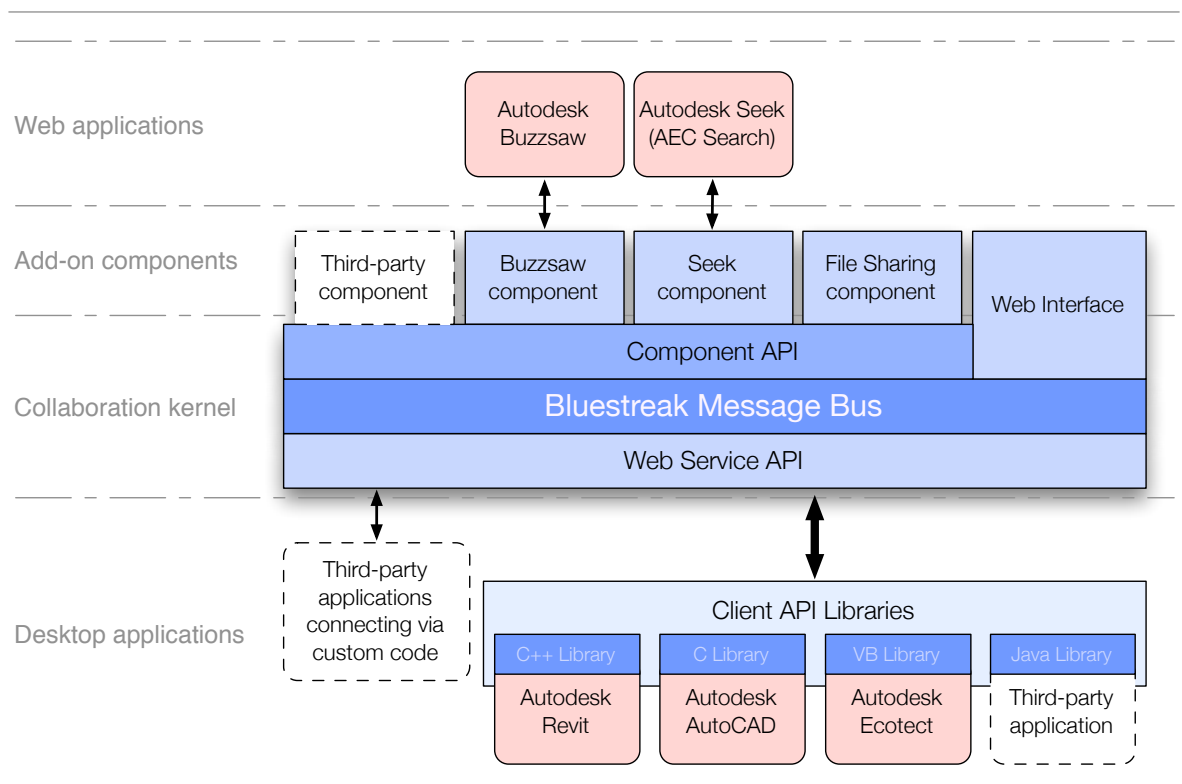
Web Service API - For basic operations many Autodesk and third-party web applications will interact with Bluestreak using a set of web service functions. Web services are a

ubiquitous and accessible means of exchanging data between different systems, but these same properties make them an inefficient means of programming complex tasks.

Client API Libraries - Learning a set of web services and writing custom code to interact with them poses a significant learning curve and development burden. To remove these barriers, Autodesk need to provide software libraries that allow developers to reliably and efficiently perform complex Bluestreak operations with only a few lines of code.

To improve the comprehension of developers and users, it is important that these three integration points are well designed and documented. A developer should not be expected to understand the entire Bluestreak platform if all they wish to do is achieve quick results using a Client API library. The experience of the end-user should be such that they are unaware these underlying interfaces exist. From their perspective, Bluestreak should be as transparent as possible, and collaboration across different applications should ‘just work’.

Figure 9.8: The three proposed layers of Bluestreak application integration



9.3.4.iv. Context sensitivity

As implemented, the only way Bluestreak embodied context sensitivity was in the use of groups to divide people and conversations. To act as a collaboration kernel, Bluestreak would need to make better use of the contextual information within a project, so that participants can easily navigate, filter and target collaboration interactions. For example project teams have clearly defined, hierarchical relationships that reflect the roles and expertise of each participant. A collaboration kernel that successfully leverages this knowledge will be more able to deliver timely and relevant information to the team. As implemented, users of Bluestreak had profile pages, but these lacked information that would help to bring relevant messages to user's attention, such as their expertise or fields of interest. Alternatively this information would help people locate other members of the team who are capable of resolving a specific design problem.

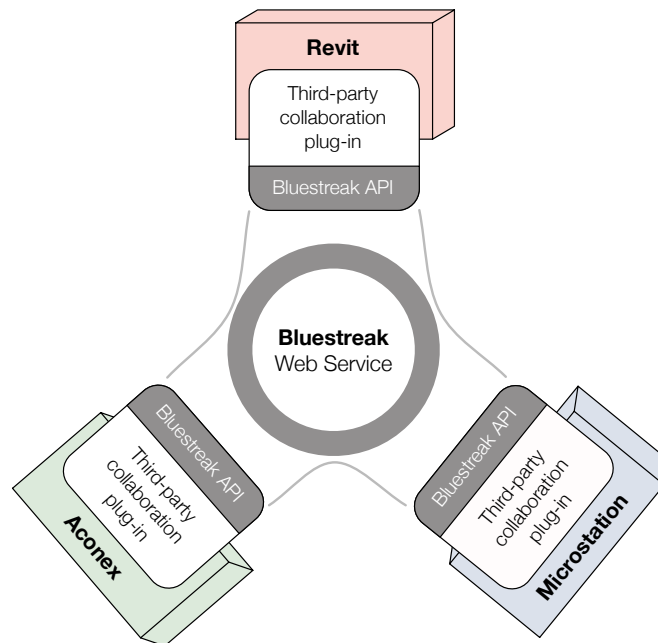
Beyond filtering and highlighting conversations, context is a useful means of stopping information from reaching participants in the first place. As it was implemented, a Bluestreak project was like working with a group of people in a large auditorium, anybody could hear or say anything. Whilst this is acceptable for general situations, when large numbers of people or sensitive data is involved it becomes important that certain interactions can occur in private. Multiple groups within Bluestreak could be created to achieve this, but this would soon become unwieldy. A more flexible approach would be if messages could be addressed to people within the team based on their profile's meta-data, or the project's hierarchical structure. This could be achieved by combining the micro-blogging conventions of the @ sign (to) and hashtag (subject, see Section 9.2.3.vi). For example, a message beginning with @#architect would signify it should be brought to the attention of architects within the team. This same mechanism could be extended to specific phases in the project (@#construction), or fields of interest (@#concept). Borrowing again from micro-blogging, a leading 'd' character (for Direct Message) would signify that the message was intended for a restricted audience. Whilst this syntax is simple, it is backwards compatible with micro-blogging standards.

9.3.4.v. Emulative modularity

Bluestreak emulates many popular social networking concepts, and the functionality it was initially released with could be easily replicated by a third-party system. Increasing the embodiment of the principle of emulative modularity within Bluestreak hinges on the

development of an API, because this will allow third-parties to extend its functionality using new components. In addition, a well documented and public API could be reimplemented by other collaboration systems, for example ProjectWise, Aconex and Vuuch, to enable components developed for Bluestreak to integrate with, or run within these other services. This modularity would have two benefits. Firstly, a highly successful application developed using the Bluestreak API would not necessarily be restricted to running exclusively within Autodesk’s collaboration environment. Secondly, this modular API would enable the collaboration kernel to act as a intermediary between applications, as illustrated in Figure 9.9. This common backbone would improve the emulative modularity of the broader digital environment, because it would allow different applications to be used without interrupting the underlying collaboration dynamic.

Figure 9.9: The collaboration kernel as a hub for integrating third-party applications



At a technical level, Bluestreak employed OpenID for user authentication, which is a modular and decentralised authentication and identity management system designed for the Web (Recordon & Reed, 2006). Unfortunately, in the initial release of Bluestreak the potential of OpenID was hamstrung because user’s were limited to Autodesk’s own OpenID provider. If Bluestreak is to act as a collaboration kernel it would need to allow third-party OpenID services to be used. This change would allow team members to participate in design conversations using digital identities they have already created within their organisation, or other third-party social networking and collaboration tools (Arrington, 2009).

9.3.4.vi. *Emotive semantics*

As implemented, Bluestreak was unable to categorise the content that was contributed by team members. When navigating and searching large quantities of collaboration data this soon becomes a problem, especially considering the content of many messages do not directly reflect its subject matter. For example, a discussion centred around ‘indoor and outdoor flow’ dealing with a conceptual idea such as a floor layout, or a specific issue such as the detailing of a door. Micro-blogging services like Twitter have demonstrated that semantics can be embedded within messages via hashtags (see Section 9.2.3.vi). Bluestreak could support a similar semantic mechanism, and components developed using its API would allow the project’s semantic structure to be visualised and navigated.

Embedding hash tags within messages is a flexible means of publishing semantics, but participants must also be able to retrospectively apply meaning to content. For example, a project’s taxonomy will initially focus on conceptual ideas, but as the design is refined, so too will the semantics used to describe it. Semantics can also have a different meaning and value depending on the perspective of the participant. For example, the various professions involved in the project may use terminology and methods of categorisation that would be unfamiliar to other parties. To compensate, a collaboration kernel should allow multiple semantic layers to be assigned to content. Achieving this semantic flexibility would require users to possess the ability to manually re-categorise content. In addition, the collaboration kernel itself should be capable of inferring meaning based on a message’s context.

Applied semantics - Users should be able to assign tags to any contributed content within Bluestreak so that it can be easily referenced. In a distributed environment embedding new semantic information within existing content is problematic, because these changes must be replicated throughout the team. An efficient means of addressing this problem is to assign all content published to Bluestreak a globally unique URL. These simple URL references could then be categorised multiple times using an existing bookmarking and tagging services such as Delicious (Halpin, Robu & Shepard, 2006, p. 3).

Inferred semantics - Beyond manual tagging, semi-intelligent agents could categorise collaboration data based on where and when it was created and what it is related to. This would require that Bluestreak was integrated into other software, so that tags could be automatically included based on information within this environment. For example, an architect using Revit may identify and highlight an issue with the design. On posting the

issue to Bluestreak using a tool built into Revit, relevant meta-data such as the components affected, materials used, and the model's revision details could be automatically included.

9.3.4.vii. Decentralisation

Like many web-based social networking applications, as Bluestreak was initially implemented it could not be installed onto a private server. Whilst this may suffice for a consumer application, it poses adoption barriers within the AEC industry, because organisations require reliable systems that adhere to entrenched processes and policies. Given these requirements, a successful collaboration kernel must be decentralised in a manner that allows it to also be deployed within organisations and integrated with existing systems.

The first step in this process would be to offer Bluestreak as a standalone application that could be installed onto a local server. Although this sounds straightforward, in practice it would require significant changes to the way Bluestreak is designed and implemented. An isolated Bluestreak server is of limited value if it cannot 'talk' to Bluestreak installations that are deployed within other organisations. In an ideal environment, each organisation should be able to operate their own Bluestreak collaboration kernel, and the messages that are published to each should be automatically delivered to other Bluestreak servers associated with the project. Unfortunately, enabling this level of reliable and timely data exchange is fraught with technical and organisational challenges. One potential solution would be for Bluestreak to implement the Wave Federation Protocol from Google (Sire, Bogdanov, Palmér & Gillet, 2009, p. 10). This protocol defines how a series of distributed Wave-compatible servers can exchange collaboration-related information in near real-time. Such functionality would enable messages that were passed to the collaboration kernel to be immediately broadcast throughout the distributed project team.

9.3.5. Summarising the collaboration kernel

As proposed in this research, a collaboration kernel could facilitate the communication of key design ideas, issues and decisions between the disparate digital collaboration tools used within the AEC industry. By helping to weave together these various communication channels, the collaboration kernel would improve the timeliness and relevancy of information delivered to members of the project team, and improve their ability to collaborate and reflect upon the design process undertaken. Applying the principles of Hyperlinked Practice in the design of a collaboration kernel exposed its key functional characteristics, their influence on the flow of information within the team, and what effect this would have on the recording of the design process.

9.4. Summary

The process of applying the principles of Hyperlinked Practice during these thought experiments demonstrated that they are a useful means of reviewing the performance of existing digital collaboration tools. The principles helped to identify specific functional areas where these digital tools could be improved so that they could more appropriately facilitate Hyperlinked Practice and record the design process. The outcomes of these thought experiments also served as an illustration of how a collaboration environment built on the principles of Hyperlinked Practice would function, and what Web-centric technologies could enable it. The outcomes of this process are an indication that a collaboration environment that embodies the seven identified principles would be capable of facilitating Hyperlinked Practice.

10. Discussion and Summary

A review of the research findings, their industry applicability, and future work

This research has identified how information technology can be developed to assist architectural project teams to more comprehensively record and reflect upon the design process and improve collaboration. Recording the design process enables more effective collaboration by promoting understanding within the project team. The current trend of consolidating project information within complex Building Information Models is not conducive to this understanding, because the majority of the team cannot access or interact with these resources. To address this problem, it is proposed that the industry adopt Hyperlinked Practice, which is the creation of a distributed cloud of interconnected information describing the project's events, activities and digital artefacts.

This research has identified seven principles that can be used to guide the design and deployment of a digital collaboration environment that facilitates Hyperlinked Practice. These principles were derived from concepts proven within the World Wide Web, which is the largest and most successful medium for publishing and interacting with digital information. To validate these principles, their collaboration influence, potential, and industry applicability was tested within a software prototype utilised in a university architecture course and two thought experiments. Overall, the results from these tests indicated that the principles positively influenced collaboration, and were valuable in guiding the design of digital tools that supported Hyperlinked Practice. However, further research is required to validate the influence and applicability of these principles within professional project teams. Depending on the research approach and resources available, future work could take two different directions. Thoroughly testing each principle separately in a controlled manner would provide a clearer picture of the specific nature and collaboration influence attributable to each principle, but this focused, "reductionist" approach would ignore how the principles work together to record the design process and influence collaboration. In contrast, testing a software prototype that embodied all of the principles within a professional project team would provide information as to whether the principles of Hyperlinked Practice work together to achieve the desired outcome in a 'real world' setting, but it would be a complex and resource intensive undertaking.

10.1. Recording the design process through Hyperlinked Practice

Effective architectural collaboration is important because each project requires a team to solve a unique design problem that has tight financial and time constraints. Accessing timely and relevant information during a project tends to be difficult because the assembled team is comprised of many geographically distributed participants from different organisations and professions. The complexity of this environment, coupled with the quantity of information exchanged, impedes an individual's ability to maintain a strong understanding of the ongoing design process, its issues, and the relevant parties who should be involved in discussions. Creating a digital record of this architectural design process is important, because it allows a Building Story to be constructed that describes the project's key events, and the activities and artefacts that shaped them. Previous research has demonstrated that these Building Stories are valuable collaboration and learning aids, because they allow participants to reflect upon a project's history, its design issues, and the personalities involved. This knowledge promotes a better understanding of the project, which allows participants to access and contribute more timely and relevant information.

During the later half of the 20th Century, architectural collaboration underwent a rapid digital transformation as many of its processes were computerised, and organisations were linked by high-speed Internet connections. These technology driven changes have overcome many longstanding architectural collaboration barriers, and now project teams can frequently, and almost instantaneously, exchange large quantities of information over any distance. These changes have revolutionised the way project teams operate, but they have also posed new collaboration challenges. Many teams are now struggling to process the large quantities of digital information that is frequently exchanged by email and within digital models. In addition, the inconsistent adoption and knowledge of information technology within project teams has created further communication barriers, and reduced the effectiveness of many collaboration initiatives. This digital fragmentation has led to a situation where team members are often unable to participate in conversations or comprehend the project's design process, because the relevant information is digitally recorded in locations and formats they cannot access.

To meet these digital collaboration challenges, the Architecture, Engineering and Construction (AEC) industry is rapidly adopting Building Information Model (BIM) technologies that enable project information to be consolidated and centrally managed

within a single, semantically rich, digital model. Unfortunately, BIM is not a suitable approach for recording or reflecting upon the architectural design process, because centralising and tightly controlling information tends to create a complex system where most of the information passing in to, or out of a BIM is handled by a relatively small number of participants. This has exacerbated the team's digital fragmentation as these access limitations and the rigid data-structure of BIM, promote the filtered and editorialised recording of project events, issues, decisions and design outcomes. BIM servers may relieve some of these access concerns, but numerous technical and practical challenges must be overcome before widespread adoption of this technology will occur.

Digitally recording the architectural design process requires an inclusive and flexible approach that ensures all team members can contribute information and reflect upon the project's Building Story. The World Wide Web is a model for such an environment, because it is the largest and most successful medium for the consumption and publishing of digital information. The characteristics that have made the Web successful are the antithesis of BIM. Instead of consolidating data within a few highly structured and controlled locations, the Web is a distributed resource that is comprised of many small pieces of information, loosely connected by hyperlinks. There are few barriers to participation on the Web given the broad range of software that is compatible with its foundational technologies, HyperText Transfer Protocol (HTTP) and HyperText Markup Language (HTML). Although there have been numerous initiatives to utilise web technologies within the AEC industry, these have generally been conservative extensions of traditional processes. For example, project intranets are commonplace within many teams, but the majority are simply centralised and tightly controlled digital filing cabinets for models and other design artefacts. Instead of taking these technologies on face value, the AEC industry needs to embrace the Web's underlying lessons, and evolve its digital collaboration environment so that the design process can be comprehensively recorded.

To record and reflect upon the design process, it is proposed that architecture teams adopt Hyperlinked Practice. This concept argues that project information should be perceived as a distributed, nebulous cloud of interconnected data, much like the Web. Rather than consolidating all project knowledge within highly structured Building Information Models (BIM), the Web has illustrated that a more inclusive approach can be achieved if participants publish, and link to smaller, less complex pieces of information that can be easily accessed. As is the case with the Web, ensuring that all parties can contribute equally

to this cloud of information is important, because this will increase the likelihood that the resulting Building Story will be an accurate and comprehensive depiction of the design process. Technologies such as BIM will remain a key resource within architectural projects. However, Hyperlinked Practice will provide a flexible and accessible means of describing the role of these digital artefacts, the events which shaped them, and their influence on the design process. Tools embodying the principles of Hyperlinked Practice will allow participants to contribute to, navigate and organise this cloud of project information in ways that best meet their specific needs.

Hyperlinked Practice is not a single technology but a change in the way the industry perceives, organises and utilises information related to architectural projects. Facilitating Hyperlinked Practice is therefore a case of gradually reshaping the industry's digital tools and collaboration processes. To achieve this long-term goal, this research has identified seven fundamental principles of Hyperlinked Practice these principles were developed by reviewing the successful characteristics of the Web, and applying them to the challenges faced during the recording of the architectural design process. These seven principles are:

- **Situational awareness** - Digital collaboration tools should integrate into the surrounding environment, so that changes that may affect the project are automatically recorded and presented to the team.
- **Ubiquity** - The digital collaboration environment should be based on commonly used processes and technologies, so that any team member may access or contribute to the project's digital record.
- **Comprehension** - All team members should be capable of understanding the purpose, implementation and operation of the project's digital collaboration tools, so that they can appropriately use them in the recording of the design process.
- **Context sensitivity** - Digital collaboration tools should understand and reflect the organisation and current state of the project, so that team members are presented with information that is relevant to the design process and their role within it.
- **Emulative modularity** - The recording and recalling of the design process should not depend on a specific technology or party. Therefore, the digital collaboration environment should be capable of being reproduced or extended by a third-party.

- **Emotive semantics** - The digital collaboration environment should not dictate the semantic system used to record or reflect upon the design process. Instead, the team should be able to define a vocabulary that reflects the uniqueness of each project.
- **Decentralisation** - The digital collaboration environment should respect the team's distributed nature and broad requirements, by not demanding that the design process be recorded in a location that is difficult to access, or controlled by a single party.

A software prototype and two thought experiments were used to validate the influence of these principles and the ability of Hyperlinked Practice to facilitate the recording of the design process. The collaboration influence of the principles was demonstrated through the testing of the Reasonate software prototype within a collaborative digital modelling class at Victoria University of Wellington. The principles were used to identify functional aspects of the software prototype, and during testing the influence of these characteristics was recorded. This information was used to determine if applying the principles achieved the desired collaboration effects, and if the resulting process supported the recording of the design process. Although the software prototype was a strong demonstration of the principles' practical collaboration influence, the limited time and resources available restricted how comprehensively the overall concept of Hyperlinked Practice could be tested. To provide a broader demonstration of the long-term potential of Hyperlinked Practice, the principles were used to evaluate two promising digital collaboration concepts, micro-blogging and social networking. These thought experiments illustrated how the principles could be used to create a new generation of digital tools that supported Hyperlinked Practice and the recording of the architectural design process.

10.2.Key research findings

This research identified seven principles of Hyperlinked Practice that if embodied within digital collaboration tools would facilitate the comprehensive recording of the architectural design process. To validate the principles' collaboration influence, industry applicability, and long-term ability to achieve Hyperlinked Practice, they were tested within a software prototype and two thought experiments. This research found that the identified principles were valid, and that when these principles were used to develop digital collaboration tools they enabled the comprehensive recording of the design process.

10.2.1. The identified principles of Hyperlinked Practice are valid

This research assessed the principles within a software prototype and two thought experiments. As described in Section 6.3, these assessments aimed to establish the influence, applicability and usefulness of each principle. The results indicated that the identified principles were valid, and each demonstrated characteristics that made it a worthy element of Hyperlinked Practice.

10.2.1.i. Influence

The results from testing the Reasonate software prototype suggest that the principles are capable of influencing collaboration in a manner that promotes the recording of the design process, and reflection upon it. A summary of these results can be found in Table 8.1.

The collaboration influence of the principles of ubiquity, comprehension, emulative modularity and decentralisation was demonstrated to be particularly strong in the prototype assessment. The web-based, blog-inspired design of Reasonate enabled it to be rapidly adopted and frequently used by the majority of the students involved. This resulted in the creation of a comprehensive digital record of the design process. Eighty-two percent of students felt that this activity improved their experience of the course, and twenty-eight percent of students believed this often led to further critical reflection. The dividing of the Reasonate software prototype into a structured blog and a freeform Web Area created a flexible and rich environment for communicating the design process (see Section 8.2.8). The use of ubiquitous and modular technologies within the prototype's design meant that a similar collaboration dynamic could be achieved a year later, when the system was emulated using a different set of software tools (see Section 8.2.6).

The collaboration influence of the principles of situational awareness and context sensitivity was less conclusively demonstrated by the software prototype test. The automated aggregation of team content within Reasonate was found to enhance context sensitivity and support situational awareness. However, the majority of team contributions occurred in clusters, and few teams leveraged tagging which was a functionality that would have enhanced the context sensitivity of the project work. In addition, the limited number of students within each team meant that the frequency of content contributions never reached levels where situational awareness functionality such as RSS and search was required by anyone except for the course coordinator (see Section 8.3.1.i).

The influence of the principle of emotive semantics was the most difficult to assess during the testing of Reasonate. This was because the tagging functionality, derived from this principle, was not used frequently or consistently by the majority of students (see Section 8.2.7). However, although tagging was not extensively used by most students, the students that did utilise this functionality created a semantic structure that was indicative of the issues and concepts within the course. Notably, a year later when the Blogger-based system was tested within the same University course, the students used its tagging functionality extensively. As described in Section 8.3.1.vi, the low uptake of tagging functionality in Reasonate may have been due to shortcomings in the user interface, and a lack of understanding by students of the immediate and long-term value of tags.

10.2.1.ii.Applicability

The complexity and resource demands of testing a prototype within a professional project team meant that a University course was used as a testing ground for the software prototype. The general success of this prototype, and the recorded collaboration influence of the principles, suggest that they would be applicable within the industry. However, whilst the University course was a reasonable testing environment, a professional project team exhibits challenges and pressures that were not present within it (see Section 7.1.1). Nevertheless, the thought experiments indicated that the principles could be successfully applied within the industry.

10.2.1.iii.Usefulness

The principles successfully guided the design of the software prototype and digital tools within the thought experiments. In both assessments the principles demonstrated an ability to create a collaboration environment that promoted Hyperlinked Practice.

The thought experiments demonstrated that the principles provided an excellent framework for evaluating a digital collaboration tool's ability to facilitate Hyperlinked Practice. In addition, applying the principles to two prominent communication trends, micro-blogging and social networking, would lead to the development of new architectural collaboration environments that would enable the design process to be more appropriately recorded and reflected upon.

10.2.2.Hyperlinked Practice can enable the digital recording of the design process

The Reasonate software prototype and the two thought experiments illustrated that collaboration tools which embody the principles of Hyperlinked Practice will result in the comprehensive and accessible recording of the design process.

10.2.2.i. The software prototype

The Reasonate software prototype was rapidly adopted by the majority of students, and they frequently described their design process throughout testing (see Section 8.2.4). This usage pattern created a comprehensive overall record that chronicled the issues faced, decisions made, and activities undertaken during the course (see Section 8.2.5). The majority of students found that utilising the prototype improved collaboration within their respective teams, and for many this process of digital reflection led to further critical thinking on the work undertaken (see Section 8.2.2). As the prototype was based on ubiquitous web technologies, it was accessible from any Internet-connected computer. This led many students to regularly contribute to and reflect upon the design process from a variety of locations (see Section 8.2.3). In addition, the prototype's ubiquitous and modular characteristics resulted in a digital environment that could be emulated by a third party without adversely affecting the collaboration dynamic (see Section 8.2.6). Although the prototype's tagging functionality was under-utilised, the semantic structure that formed was an accurate reflection of the design and collaboration issues experienced during the course (see Section 8.2.7). The prototype also illustrated that creating a digital record of the design process is only the first stage in comprehending a project's Building Story. If participants are to leverage the recorded design process in order to improve understanding and collaboration, they require a means of efficiently organising this digital history into meaningful and accessible Building Stories (see Section 8.3.2).

10.2.2.ii. The thought experiments

The principles of Hyperlinked Practice can drive the development of digital collaboration environments that promote the recording and reflecting upon of the design process. By applying the principles to two new digital collaboration concepts, micro-blogging and social networking, the thought experiments illustrated that achieving a diverse, robust and capable Hyperlinked Practice environment is an achievable goal. This process was an example of how the principles of Hyperlinked Practice can be used to compare digital collaboration tools, or identify aspects of their technical design that could be improved

upon. The results of this theoretical analysis were proposal for two new, highly capable architectural collaboration tools, that if implemented would promote Hyperlinked Practice and the recording of the design process within the industry.

10.3.Applying Hyperlinked Practice within the industry

This research identifies a strategy for improving the AEC industry's ability to digitally record and reflect upon the architectural design process. Promoting and facilitating this within a digital collaboration environment is important, because it enables the project's events, activities and artefacts to be organised into meaningful Building Stories. The existence of these stories promotes more effective collaboration, because participants have a better understanding of the issues, personalities, and decisions which contributed to a project's design outcome. The current digital architectural collaboration trend is to consolidate and distill project information into complex Building Information Models. However, due to the centralised, rigid and complex nature of this technology, it is unable to comprehensively record the design process. This research argues that the information related to the design process should instead be conceived and recorded as a distributed and interconnected cloud of data. The World Wide Web has demonstrated that this is a more flexible and inclusive approach when large numbers of people wish to record, share and reflect upon a diverse range of subjects. To provide a means of achieving a new digital collaborative process for the AEC industry, this research identified and validated the seven principles of Hyperlinked Practice. These principles are significant because they apply the core lessons of the Web to the collaboration challenges currently faced within architecture teams. Thought leaders, software developers, and industry decision makers can leverage these principles and associated research in a number of ways:

Thought leaders - This research is an initial demonstration of how Building Stories can be digitally recorded and utilised within project teams. As described in Section 2.4.1, previous research by Martin, Heylighen and Cavallin (Martin *et al.*, 2003) conceived of Building Stories, but the processes they used to create these resources were impractical within a typical architectural project team. Hyperlinked Practice and its principles provides an initial indication of how Building Stories can be realised and utilised within project teams. This research forms a strong theoretical foundation, which future technologies and software products can build upon to record and leverage Building Stories within project teams. In addition, Hyperlinked Practice is a critique of the current trend towards

consolidation of design knowledge within Building Information Models. Given the digital fragmentation of project teams, this research has shown that alone BIM cannot adequately record the design process, or achieve its full collaboration potential. Industry thought leaders can use Hyperlinked Practice and its principles as a means of overcoming these shortcomings. The two approaches are not mutually exclusive, and a BIM that was enveloped by and integrated into a project information cloud would be a powerful combination of technologies.

Software developers - As illustrated in the thought experiments of Section 9, the principles can be used to guide the development of digital collaboration tools that allow the design process to be more comprehensively recorded and reflected upon. From the perspective of software developers, the overarching concept of Hyperlinked Practice promotes a diverse and active collaboration ecosystem. As discussed in Section 9.3.2, this environment would benefit all involved, because it would expose project information in ways that are currently not possible due to the insular and isolated nature of most digital collaboration processes. Individually, each principle describes how functional changes within a digital tool can influence specific aspects of collaboration. Thus, software developers could apply the process used for developing the Reasonate software prototype (as described in Section 7.2), to identify how specific aspects of the team's collaboration dynamic could be improved through targeted functional changes to existing digital tools.

Industry decision makers - Currently there are few decision making aids available for guiding or justifying the procurement and implementation of digital collaboration technologies. As a consequence, the application of information technology within the industry has been inconsistent, which has led to the digital fragmentation of many project teams (see Section 3.1.3.iii). The principles of Hyperlinked Practice provide industry decision makers, such as directors, managers, and IT consultants, with a framework for reviewing and comparing the potential benefits and drawbacks of different digital collaboration tools and strategies. As illustrated within the thought experiments in Section 9, the principles provide seven different axes for reviewing and comparing performance. To further assist decision makers in their analyses of digital tools, future research could determine whether relationships exist between how thoroughly a digital tool embodies a principle, and the collaboration influence that is recorded. This would be valuable knowledge, as it could assist in providing an evidence based answer to more detailed questions such as, 'if System A is twice as distributed as System B, will this yield twice the

collaboration benefit?’ Definitive and quantified answers to these questions are not possible owing to technical and implementation differences, but historical case studies and controlled experiments could allow generalised observations to be confidently made.

Through their choices and actions, industry decision makers could play a key role in promoting Hyperlinked Practice, and facilitating a digital environment where project information is treated as a distributed, accessible and malleable virtual cloud of data. Transitioning from the isolated and disparate digital collaboration environment of today will take time, but through persistent and consistent application of the principles of Hyperlinked Practice, appropriate technologies, tools, and deployment strategies will be developed and applied. In this eventuality, existing collaboration technologies such as BIM, CAD and email will not be replaced, but the functionality of these tools will need to evolve so that they can support Hyperlinked Practice, and form an active part of a project’s information cloud.

10.4.Limitations of this research

The inherent difficulties of conducting experiments related to collaboration within architectural project teams made validating the principles of Hyperlinked Practice a challenging process (see Section 6.2). The limited time and resources available for the research meant that the tests were an initial exploration of the validity of the principles, and that further tests are required to fully validate the principles. Although the applicability of the principles to a professional project team can be inferred from the tests on student project teams and the thought experiments, it would be beneficial to be able to demonstrate this applicability more directly. Consequently, direct testing of the principles within a professional project team context should be a focus of future research.

10.4.1.The design and testing of the Reasonate software prototype

Testing the Reasonate software prototype within a professional project team would have provided the strongest indication of the principles’ collaboration influence and industry applicability, but the demands of this environment exceeded the time and resources available for this research. The BBS303 ‘Digital Craft’ University course was an appropriate and practical testing environment, but the close working proximity of the students was a limitation considering the research theme of digital collaboration. All of the

students worked within the same building, which led to the majority of the collaboration interactions occurring in person. As a consequence, Reasonate was primarily used to record the decisions and progress made. This meant some of the prototype's collaboration-specific functionality was under-utilised, because students could physically see their teammates progress, and request information from them in person. In future research, a more appropriate testing environment would be one where the team is geographically distributed. For example, in the case of Reasonate its project teams could have been comprised of students from different campuses or Universities. Although this imposes a greater research burden and risk, it would lead to an environment that is more indicative of industry conditions.

The design of the Reasonate software prototype was a complex balance between testing the collaboration influence of the principles, the needs of the students within the course, and the time and resources available for development. In some cases this led to principles not being tested as thoroughly as what would be considered ideal. If more resources had been available, a multivariate testing strategy would have been a more comprehensive way of demonstrating the collaboration influence of specific principles (Tractinsky, Katz & Ikar, 2000, p. 136). Having different groups within the class use functionally different versions of Reasonate would have been a more robust means of measuring the influence of the principles on the prototype's operation. Unfortunately, considering the time and resources required to develop one version of Reasonate, it was impractical to consider testing multiple, functionally different versions during this research.

The analysis of the Reasonate test results in Section 7.3 highlighted three key limitations in the design of the software prototype and the methods used to collect data during testing. These limitations may have influenced or impaired the assessment of the principles' recorded influence:

- **Issues with the design of the user interface**

Principles affected: emotive semantics and context sensitivity.

As described in Sections 7.4.3 and 8.3.1, there were several instances where Reasonate's user interface had an undue influence on the recorded collaboration effects. Many of these issues were addressed during testing, but often they had already permanently influenced the student's perception and usage patterns.

- **The frequency of the usage questionnaire**

Principles affected: situational awareness, comprehension, emulative modularity.

As described in Section 8.3.3.i, questioning students on their Reasonate use once at the end of testing was a limitation. Students should have been asked to complete the online questionnaire at regular times during testing. This would have strengthened the analysis of the principles' influence, and helped to identify functional aspects of the prototype that were not being utilised as intended.

- **Analysis of web access analytics**

Principles affected: comprehension and decentralisation.

Daily statistics on the content created in Reasonate were available for analysis, but data associated with how often students accessed this data was not. As described in Section 8.3.1.vii, this information would have provided more insight into the adoption and content consumption patterns of students.

Addressing these areas and retesting the software prototype would have potentially led to results that were of a higher quantity and quality. Unfortunately, given the yearly scheduling of the course and the time required to organise and conduct the experiment, repeating the experiment was impractical.

10.4.2.Thought experiments

The thought experiments in Section 9 were a strong illustration of how the principles could be applied to emerging digital collaboration technologies, and how this would promote Hyperlinked Practice and the recording of the design process. However, this discussion was purely theoretical, and the outcomes reached were not directly tested within the industry. In future research, grounding within the industry could be achieved through the use of case studies or discussion groups. For example, Burry, Burrow and Burry (2005) reviewed the historical email correspondence of an architectural project to determine what information was exchanged and how it could be recorded within a wiki. A similar approach could be applied in order to ground the micro-blogging and social networking thought experiments. However, it is questionable whether this exact approach would have strengthened the thought experiments, because email exchanges are very different in nature to the interactions that occur within micro-blogging and social networking environments (Grippa, Zilli, Laubacher & Gloor, 2006). As a result, whilst case studies and discussion groups may strengthen a theoretical argument, validating the principles of Hyperlinked Practice ultimately requires practical testing under industry conditions.

10.4.3.The organisational and legal barriers to Hyperlinked Practice

This research intentionally limited its scope to the information technology issues and requirements for recording and reflecting upon the design process. It is acknowledged that there are numerous organisational and legal barriers that may restrict the recording of this information. Once the technical validity and applicability of Hyperlinked Practice has been established, the focus of future work would shift to studying these barriers and identifying appropriate means of overcoming them.

10.5.Future work

This research was an initial exploration of the concept of Hyperlinked Practice and the ability of its underlying principles to promote the digital recording and reflection of the design process. As discussed in Section 10.4, time and resource limitations, coupled with the difficulties of undertaking research within professional project teams, means that a considerable amount of future work is required in order to thoroughly validate the principles of Hyperlinked Practice. To achieve this goal, future work could take two different lines of inquiry depending on the resources available, and whether the researcher adopted a positivist or interpretative research paradigm (Davidson & Tolich, 1999, p. 26). A positivist line of investigation would be more likely to favour a reductionist approach focusing on isolating, validating and quantifying the effects of each principle, whereas an interpretative approach would emphasise the collective testing of the principles within a professional project team. Both of these lines of inquiry are valid, and would further demonstrate the feasibility and collaboration benefits of Hyperlinked Practice. This knowledge would be valuable in establishing the business case for Hyperlinked Practice, and identifying the potential organisational and legal barriers that could influence adoption.

10.5.1.The isolated validation of each Hyperlinked Practice principle

Due to time and resource limitations, the Reasonate software prototype did not definitively validate the influence of the principles, or quantify their collaboration influence (see Section 10.4). Future research could address these limitations by employing a reductionist research approach to comprehensively validate and understand the effects of each principle. Undertaking this research would strengthen the foundations of Hyperlinked Practice, and provide decision makers with more insight as to how the principles could be used to guide the development of digital tools and strategies.

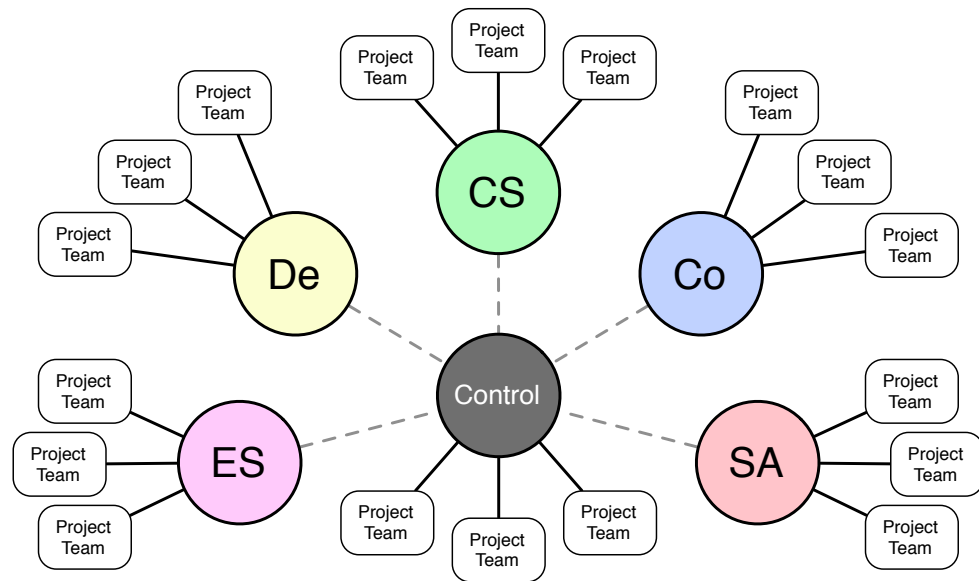
A relatively straightforward means of isolating and studying each principle in detail would be to use an A/B testing strategy (Kohavi, Henne & Sommerfield, 2007) to test multiple derivatives of a software prototype similar to Reasonate. As described in Section 10.4, testing multiple derivatives of the same underlying prototype would lead to a stronger understanding of how the functionality derived from a principle influenced collaboration and the recording of the design process. Isolating the effects of a specific functional change would require an unmodified prototype to act as a control. The modified prototype and control would need to be tested within the same testing environment, to ensure the

recorded results were comparable. Testing all seven principles using an A/B testing strategy would therefore require, at minimum, eight versions of the software prototype, one to test each principle related change, and an unmodified control (see Figure 10.1). Table 10.1 describes how the existing Reasonate prototype could be modified and tested using an A/B testing strategy to better understand the principles' influence. However, two of the principles, ubiquity and emulative modularity, could not be easily tested in this manner, because they are embodied throughout Reasonate's design. Despite this, the test results would comprehensively demonstrate the principles' collaboration influence, because direct comparisons could be drawn between the usage characteristics of the different versions of the prototype.

Table 10.1: The functional differences between the control and alternate prototypes

Principle	Control functionality	Alternate functionality
Comprehension (Co)	Functionality is gradually deployed and introduced throughout the testing period.	All functionality is deployed on the first day of testing with minimal instruction.
Context Sensitivity (CS)	Contributions of team members are automatically aggregated into a project blog.	There is no concept of project teams. Instead, users must visit team member blogs.
Decentralisation (De)	Users can access the prototype from any Internet connected computer with a web browser.	Users can only access the prototype from designated computers.
Emotive Semantics (ES)	Users can file content using a semantic vocabulary they have defined using tags.	Users can file content based on a fixed set of categories defined by a third-party.
Situational Awareness (SA)	Users can view any content that is contributed to the prototype.	Users can only view the content that they have contributed.

Figure 10.1: Applying an A/B testing strategy to the testing of Reasonate



This approach would lead to stronger conclusions on the validity and influence of the principles, but it does have a number of limitations:

Industry applicability - Testing multiple derivatives of the software prototype would be an extremely difficult process within a professional project team, because it would be difficult to control for all the potential external factors that could affect the comparability of the results. In addition, testing all the principles separately would require a large number of participants. It is likely that the only way adequate numbers could be achieved is if testing spanned multiple projects and organisations. This would likely lead to different patterns of use that would invalidate any comparisons between prototype results. Consequently, in order to minimise external confounding factors, and ensure a relatively consistent user base, it may be beneficial to undertake testing within a more controlled environment such as a University course.

Interdependency between principles - This reductionist approach would not take into account the potential relationships and dependencies that exist between the different principles. For example, one principle may only influence collaboration if it is applied in partnership with another. Testing each principle independently would not accurately reflect the importance of these relationships.

Relationship between cause and effect - The testing process described would not determine whether there was a relationship between the size of the recorded collaboration influence, and how thoroughly a principle was embodied by a digital tool. As described in

Section 10.3, forming a basic understanding of this relationship would be valuable to industry decision makers and software vendors, because they would be more able to predict the collaboration influence they could expect to see, based on a specific set of technical changes. Identifying the relationship between these two factors would be a complex and resource intensive process. Numerous software prototypes would be required that embodied a single principle to varying degrees. All of these would then need to be tested under similar conditions so that the results were comparable. This process would be extremely complicated, the prototypes would be difficult to design, and the recorded results influenced by a variety of technical issues and external factors.

10.5.2. The holistic testing of Hyperlinked Practice within the industry

The most compelling argument for the applicability and benefit of Hyperlinked Practice is a demonstration that it can promote the recording and reflection of the digital design process within a professional project team. Although the University course used to test Reasonate was a reasonable simulation of this environment, it lacked many of the complexities and pressures that make architectural collaboration demanding. To rectify this limitation, a software prototype embodying all of the principles of Hyperlinked Practice should be thoroughly tested within a professional project team. The objective of this future work would be to establish that the design process can be digitally recorded under industry conditions, and that the resulting information is valuable to the professionals involved. The benefit of this more interpretative research approach is that it would provide a strong demonstration of the applicability and value of Hyperlinked Practice within the industry. This would complement the research already undertaken, and illustrate that a digital collaboration tool that embodied the principles of Hyperlinked Practice could record the design process under industry conditions.

Designing, developing and testing a software prototype that was to be used by a professional project team is a far greater undertaking compared to Reasonate. Whilst a standalone blogging tool could be developed and tested, a more ambitious prototype would seek to implement many of the concepts discussed in the thought experiments of Section 9. During the design of this prototype, the principles of Hyperlinked Practice would be used to inform how the many practical and technical challenges faced could be overcome. Whereas Reasonate was based on a single server and only accessible through a web browser, a software prototype intended for industry use would need to be distributed

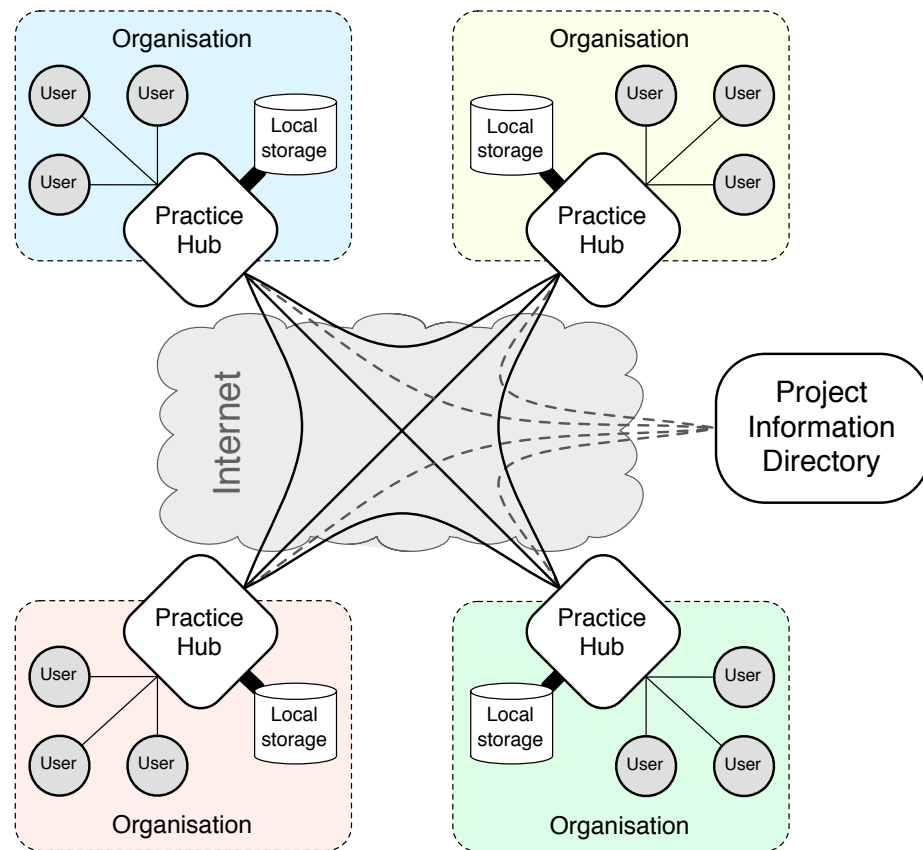
and more deeply integrated into the digital tools commonly used by professionals. The distribution of project information is a central component of Hyperlinked Practice, so an important aspect of future work would be exploring ways that team members can contribute and store project information in a distributed manner. Integrating and presenting this information within commonly used digital tools such as CAD, BIM and email is also important, because it enables the principles of situational awareness, context sensitivity and ubiquity to be more thoroughly embodied and tested. An initial, high-level design of this software prototype is illustrated in Figure 10.2. This proposal has three characteristics that were not present within the design of the Reasonate software prototype:

The Practice Hub - Each organisation would employ their own Practice Hub, which staff would interact with when contributing to, or reflecting upon, the design process. Similar to a contemporary email server, a Practice Hub would exchange project information with the other Practice Hubs that are part of the team. A Practice Hub will determine relevant recipients of design information by querying the Project Information Directory, which understands the roles and level of access of team members. This process will ensure that sensitive information is not delivered to inappropriate organisations. When a Practice Hub receives information, it notifies relevant staff and stores a copy in its archive for reference.

The Project Information Directory - Each project requires a Project Information Directory that describes the composition of the team. A Project Information Directory would operate similar to the Domain Name System (DNS) on the Internet (Mockapetris & Dunlap, 1995). This fundamental information about a project must be maintained within a single, trusted source because it defines the roles and level of access within a team. Any changes to the team's composition will be reflected within the Project Information Directory, which will allow organisations and staff to change without disrupting the flow of information. Practice Hubs will query this service to determine the relevant recipients of a message, or which staff should be notified about specific practice events.

Local Storage - Given the temporary nature of architectural project teams, there is a strong possibility that information stored by another party will not be available in the long term. In addition, team members have a tendency to communicate ideas within large digital models that are a strain on Internet connections. To compensate each Practice Hub would keep a local copy of all of the information it sends and receives, so that staff members can quickly and reliably access this information.

Figure 10.2: The design for a distributed software prototype for use within the industry



Developing and testing this prototype would follow a similar pattern to that of Reasonate. Given the complexity of the task, an agile development approach would need to be employed throughout the testing process, as this would ensure the deployed functionality was responding to the actual needs of the professionals involved. Unlike Reasonate, the professionals would be asked to complete regular online questionnaires that monitored the influence of the principles, and the performance of the prototype. To produce meaningful results, the prototype would need to be used for a relatively long period of time by many industry professionals. Testing within three or four medium sized architectural projects should provide the quantity of results needed for analysis, and a degree of redundancy in case a project is cancelled. Whereas the emphasis of Reasonate was on demonstrating the principles' collaboration influence, analysis of the industry prototype would concentrate on the design process that was recorded, and how this resource was utilised by team members during the project. The recorded data and feedback from this experiment could be compared to historical case studies and reviewed within industry discussion groups. Together these forms of analysis would generate a well rounded understanding of the prototype's influence, and the applicability of Hyperlinked Practice within the industry.

10.6.Summary

This research has shown that digitally recording the architectural design process can be achieved if the AEC industry adopts Hyperlinked Practice, which is a Web-centric means of perceiving of, communicating and storing project information. Recording the architectural design process promotes more effective collaboration within the team, because participants are more capable of organising and reflecting upon the project's Building Story. At present, comprehensively recording the design process is difficult given the digital fragmentation within the team, and the industry trend towards consolidating project information within complex and highly structured Building Information Models. To promote an inclusive digital collaboration environment, this research has identified seven fundamental principles of Hyperlinked Practice. The influence, applicability, and usefulness of these principles was tested using a software prototype and two thought experiments. The results of these tests indicated that the principles were valid, and merited further development and investigation. The most significant limitation of this research was the lack of testing within professional project teams, but given the time and resource demands of such an activity, this was deferred until future research. Despite this limitation, the research findings suggest that by adopting Hyperlinked Practice, the digital collaboration environment within the industry would be more inclusive, dynamic, and capable of recording a project's design process.

Appendix

Ethics Approval: Application and response from Victoria University of Wellington

Participant Information Sheet for Internet Design Collaboration Research

Reasonate Feedback Questionnaire

Researcher: David Harrison, School of Architecture, Victoria University of Wellington

I am a PhD student at the School of Architecture at Victoria University of Wellington. The research I am undertaking is exploring how Internet technologies can improve collaboration within the design process. During the BBS303 course you took part in last semester you were tasked with using the Reasonate application to document your team's progress, discuss issues and identify important design decisions during the duration of the course.

For my thesis research I am interested in gaining feedback from you on the effectiveness of the Reasonate tool when it came to achieving the previously described tasks and how, if at all, you felt the tool effected your understanding of the collective design and documentation process.

This questionnaire is to be completed anonymously and the collated findings will be published in the completed PhD thesis. It will not be possible for you to be identified personally as only grouped responses will be presented in the research. All material will be kept confidential and no person apart from myself and my supervisor, Michael Donn will see the submitted, uncollated questionnaires. The thesis will be submitted for marking to the School of Architecture and deposited in the University Library. The submitted questionnaires will be destroyed two years after the end of the project.

An overview of Reasonate's research intentions can be found here:

<http://www.stress-free.co.nz/content/view/271/2/>

For a more general overview of the thesis research please checkout my thesis website which has a chronological history of the thesis and its overall intentions:

<http://www.stress-free.co.nz/thesis/>

If you have further questions about the project or this questionnaire, please contact me at david.harrison@stress-free.co.nz (or phone 021 428301) or my supervisor, Michael Donn, at the School of Architecture at Victoria University, P O Box 600, Wellington or phone 04 4636221.

David Harrison

Signed:

MEMORANDUM

Phone 0-4-463 5676
Fax 0-4-463 5209
Email Allison.kirkman@vuw.ac.nz

TO	David Harrison
COPY TO	
FROM	Dr Allison Kirkman, Convener, Human Ethics Committee
DATE	
PAGES	1
SUBJECT	Ethics Approval: No 53, A digital framework for information searching and exchange within the New Zealand construction industry.

Thank you for your application for ethical approval, which has now been considered by the Standing Committee of the Human Ethics Committee.

Your application has been approved and this approval continues until 30 May 2007. If your data collection is not completed by this date you should apply to the Human Ethics Committee for an extension to this approval.

Best wishes with the research.

Allison Kirkman
Convener

Bibliography

- 37Signals (2010). *Getting real: The smarter, faster, easier way to build a successful web application*. Chicago, USA: 37Signals LLC.
- Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). *Agile software development methods*. Finland: VTT Publications.
- Adachi, Y. (2001, December 19). *Introduction of IFC model server*. Finland: SECOM Co., Ltd. / VTT Building and Transport. Retrieved May 3, 2010, from <http://cic.vtt.fi/projects/ifcsvr/memo/VTT-MEMO-ADA-05.pdf>
- Alexander, J., Coble, R., Crawford, J., Drogemuller, R., & Newton, P. (1998). Information and communication in construction: Closing the loop. In *Cib W78*. International Council for Research and Innovation in Building and Construction.
- Ambrose, M. A. (2006). Plan is dead: To BIM or not to BIM, that is the question . In *Computing in architecture / re-thinking the discourse: The second international conference of the arab society for computer aided architectural design (ASCAAD 2006)*. Sharjah, United Arab Emirates .
- Amor, R. & Faraj, I. (2001). Misconceptions about integrated project databases. *Journal of Information Technology in Construction*, 6, 57-68.
- Amor, R., Jiang, Y., & Chen, X. (2007). BIM in 2007 – are we there yet? In *24Th W78 conference "bringing ITC knowledge to work"*. Maribor, Slovenia: International Council for Research and Innovation in Building and Construction.
- Angulo, A., Fillwalk, J., & Vásquez de Velasco, G. (2009). Collaborating in a virtual architectural environment: The las americas virtual design studio (LAVDS) populates second life. In *Sigradi 2009 - proceedings of the 13th congress of the iberoamerican society of digital graphics*. Sao Paulo, Brazil.
- Anumba, C. J. (1998). Industry uptake of construction IT innovations - key elements of a proactive strategy. In *Papers from the CIB W78 workshop*. Stockholm, Sweden: International Council for Research and Innovation in Building and Construction.
- Aranda-Mena, G., Crawford, J., Chevez, A., & Froese, T. (2008). Building information modelling demystified: Does it make business sense to adopt BIM? *24Th W78 Conference "Improving the Management of Construction Projects Through IT Adoption"*.
- Arjun, G. & Plume, J. (2006). Collaborative architectural design as a reflective conversation: An agent facilitated system to support collaborative conceptual design. In *Computing in architecture / re-thinking the discourse: The second international conference of the arab society for computer aided architectural design*. Sharjah, United Arab Emirates .

- Arrington, M. (2009, April 27). Facebook first big site to really embrace openid. *Techcrunch* [Web page]. Retrieved July 7, 2010, from <http://techcrunch.com/2009/04/27/facebook-first-big-site-to-really-embrace-openid/>
- Arrow, J. (2008). RISK. 01 Knowledge-based proactive project risk management. [PDF]. Retrieved October 10, 2011, from <http://www.aacei-hgcs.org/PastNewsLetters/Knowledge-Based%20Proactive%20Project%20Risk%20Management.pdf>
- Asite (2006). *BAA save 2% on construction costs through the use of asite*. Retrieved February 10, 2010, from https://www.asite.com/images/uploads/library/case-studies/pdfs/Case_Study_BAA.pdf
- Autodesk (2007). *Profitable "paperless" building project* [PDF]. Retrieved February 20, 2010, from http://images.autodesk.com/emea_nw_w_main/files/jonsson_and_worren_case_study.pdf
- Badger (1999, June 30). *How to implement research wisdom to change the construction industry*. Arizona, USA: College of Engineering and Applied Sciences, Arizona State University. Retrieved June 23, 2009, from <http://www.ce.berkeley.edu/~tommelein/CEMworkshop/Badger.pdf>
- Barrett, P. & Stanley, C. (1999). *Better construction briefing*. Blackwell Science.
- Barrett, P., Hudson, C., & Stanley, C. (1996). Is briefing innovation? *Managing Construction Information, The Organization and Management of Construction, Shaping Theory and Practice, Volume 3*, 87-95.
- Barrow, L. (2004). Elitism, IT and the modern architect opportunity or dilemma. In *Thresholds - design, research, education and practice, in the space between the physical and the virtual - proceedings of the 2002 annual conference of the association for computer aided design in architecture*. California, USA.
- Becerik, B. (2004). A review on past, present and future of web based project management & collaboration tools and their adoption by the US AEC industry. *International Journal of IT in Architecture Engineering and Construction*, 2, 233-248. Retrieved October 10, 2009, from the Google Scholar database, http://www.burcinbecerik.com/wp-content/uploads/2008/07/past-present-future-of-extranets_intjournal-it-aec_oct-20041.pdf
- Becerik-Gerber, B. & Rice, S. (2009, September 1). The value of building information modeling: Can we measure the ROI of BIM? *Aecbytes* [Web page]. Retrieved September 4, 2009, from http://aecbytes.com/viewpoint/2009/issue_47.html
- Beck, K. (2001). *Extreme programming explained: Embrace change* (7 ed.). Reading, Massachusetts, USA: Addison-Wesley. Retrieved July 29, 2009, from the Google Scholar database, http://books.google.com/books?id=G8EL4H4vf7UC&dq=extreme+programming+explained&source=gbs_navlinks_s

- Bentley, K. (2003, January). Does the building industry really need to start over? Bentley Systems Incorporated. Retrieved June 6, 2009, from ftp://ftp2.bentley.com/dist/collateral/BIM_20White_20Paper_2C_20Low_20Respdflo.pdf
- Berners-Lee, T. & Cailliau, R. (1990). Worldwideweb: Proposal for a hypertext project. *European Particle Physics Laboratory (CERN)*. Retrieved July 7, 2010, from <http://www.w3.org/Proposal>
- Berners-Lee, T. (1996, December). *Universal resource identifiers*. Retrieved January 6, 2006, from <http://www.w3.org/DesignIssues/Axioms.html>
- Berners-Lee, T. (1998). *Principles of design*. Retrieved February 2, 2006, from <http://www.w3.org/DesignIssues/Principles.html>
- Berners-Lee, T. (1998, September). *Web architecture from 50,000 feet*. Retrieved February 3, 2006, from <http://www.w3.org/DesignIssues/Architecture.html>
- Berners-Lee, T. (2006). The world wide web-past, present and future. *Journal of Digital Information, 1*(1). Retrieved July 7, 2010, from <https://journals.tdl.org/jodi/article/viewArticle/3/3>
- Berridge, P. & Brown, A. (2002). A touring machine. In *Connecting the real and the virtual - design e-ducation 20th ecaade conference proceedings*. Warsaw, Poland.
- Beyh, S. & Kagioglou, M. (2004). Construction sites communications towards the integration of IP telephony. *Electronic Journal of Information Technology in Construction, 9*, 325-344.
- Björk, B. (1995, December). *Requirements and information structures for building product data models*. Thesis, Espoo, Finland: Helsinki University of Technology.
- Bosworth, A. (2004, November 18). ISCO04 talk. *Adam bosworth's weblog* [Web page]. Retrieved May 6, 2009, from <http://adambosworth.net/2004/11/18/iscoc04-talk/>
- Bosworth, A. (2005, April 18). *Database requirements in the age of scalable services*. O'Reilly Media MySQL Conference. MP3 from IT Conversations. Retrieved October 25, 2009, from <http://itc.conversationsnetwork.org/shows/detail571.html>
- Böhringer, M. & Richter, A. (2009). Adopting enterprise 2.0: A case study on microblogging. In *Mensch & computer 2009: Grenzenlos frei!?* Munich, Germany: Oldenbourg Verlag.
- Buckheit, J. & Donoho, D. L. (1995). Wavelab and reproducible research. *Stanford university* [Web page]. Stanford, California, USA: Stanford University. Retrieved June 7, 2009, from the Google Scholar database, <http://www-stat.stanford.edu/~donoho/Reports/1995/wavelab.pdf>

- Burry, J., Burrow, A., & Burry, M. (2005). Upholding the poetic in design collaboration. In *CAADRIA*. New Delhi, India.
- Burry, J., Burrow, A., Amor, R., & Burry, M. (2005). Shared design space. In *Proceedings of the 11th international conference on computer aided architectural design futures*. Vienna, Austria.
- Cameron, K. (2005). The laws of identity. *Kim cameron's identity weblog* [Web page]. Retrieved July 7, 2010, from <http://www.identityblog.com/stories/2004/12/09/thelaws.html>
- Casey, D. (2010, April 14). Replay it: Google search across the twitter archive. *The office Google Blog* [Web page]. Google. Retrieved June 5, 2010, from <http://googleblog.blogspot.com/2010/04/replay-it-google-search-across-twitter.html>
- Cayzer, S. (2004). Semantic blogging and decentralized knowledge management. *Communications of the ACM, Vol 47, No 12*, 47-52.
- Chen, Y. Z. & Maver, T. W. (1996). Supporting interaction within virtual studios. In *W78: Construction on the information highway*. Bled, Slovenia.
- Chrabin, A. M., Neuckermans, H., & Szewczyk, J. (2004). A critical evaluation of early stages software in its capacity of coping with contextual issues. *Local Values in a Networked Design World: Added Value of Computer-Aided Architectural Design*, 165-181.
- Christiansson, P. (1996). Knowledge communication in the building industry. The knowledge node concept. In *Cib W78*. International Council for Research and Innovation in Building and Construction.
- Christiansson, P. (1998). Using knowledge nodes for knowledge discovery and collaboration. In I. Smith (Ed.), *Artificial intelligence in structural engineering*. Springer, Berlin.
- Christiansson, P., Da Dalto, L., Skjaerbaek, J., Soubra, S., & Marache, M. (2002). Virtual environments for the AEC sector – the diversity experience . In *ECPPM 2002: Ework and ebusiness in AEC*. Slovenia..
- Cooper, G., Cerulli, C., Peng, C., & Rezgui, Y. (2005). Tracking decision-making during architectural design. *Itcon*, 125-139.
- Davenport, T. H. & Pearlson, K. (1998). Two cheers for the virtual office. *Sloan Management Review*, 39, 51-66.
- Davidson, C. & Tolich, M. (2003). *Social science research in new zealand: Many paths to understanding* (2 ed.). Auckland, New Zealand: Pearson Education Limited. (Original work published 1999)

- Denning, S. (2004). Telling tales. *Harvard Business Review*, 82(5), 122-129.
- Domeshek, E. & Kolodner, J. (1993). Using the points of large cases. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 7(2), 87-96.
- Dougherty, D., Kimber, E., Rizk, A., Russel, D., & Summers, K. (1994). Panel: HTML: Poison or panacea? In *ECHT'94 proceedings*.
- Eastman, C. M., Teicholz, P., Sacks, R., & Liston, K. (2008). *BIM handbook: A guide to building information modeling for owners, managers, architects, engineers, contractors, and fabricators*. New Jersey, USA: John Wiley and Sons.
- Edmunds, A. & Morris, A. (2000). The problem of information overload in business organisations: A review of the literature. *International Journal of Information Management*, 20(1), 17-28.
- Egan, J. (1998). *Rethinking construction: The report of the construction task force*. Department of Trade and Industry, London, UK: The Construction Task Force.
- Emmitt, S. & Gorse, C. (2003). *Construction communication*. Oxford, UK: Blackwell Publishing Ltd. Retrieved July 2, 2009, from the Google Books database, <http://books.google.com/books?id=KwMLU-QVj2sC&d>
- Engsbo, M. (2003). Barriers to adopting internet-enabled collaboration and business in the building construction industry-towards a three-level framework. *Frontiers of E-Business Research*, 456-462.
- Erenkrantz, J. & Taylor, R. (2003). Supporting distributed and decentralized projects: Drawing lessons from the open source community. In *Proceedings of 1st workshop on open source in an industrial context*. Anaheim, California. Retrieved August 30, 2009, from <http://www.erenkrantz.com/Geeks/Research/Publications/Open-Source-Process-OSIC.pdf>
- Evbuomwan, N. F. O. & Anumba, C. J. (1998). An integrated framework for concurrent life-cycle design and construction. *Advances in Engineering Software*, 29(7-9), 587-597.
- Falkowski, G. & Troutman, S. (2005). *Remote control: A practitioner's guide to managing virtual teams*. Texas, USA: Rector Duncan & Associates. Retrieved April 5, 2010, from the Google Books database, <http://books.google.co.nz/books?id=CQK1--YbQUEC>
- Farhi, P. (2009). The twitter explosion. *American Journalism Review*, 31(3), 26-31.
- Fielding, R. & Taylor, R. (2002). Principled design of the modern web architecture. *ACM Transactions on Internet Technology (TOIT)*, 2(2), 115-150.
- Froese, T., Fischer, M., Grobler, F., Ritzenhaler, J., & Yu, K. (1999). Industry foundation classes for project management - A trial implementation. *Itcon*.

- Froese, T., Grobler, F., & Yu, K. (1998). Developing data standards for construction - an IAI perspective. In *W78: The life-cycle of construction IT innovations. - technology transfer from research to practice*. Stockholm, Sweden.
- Garrison, D. R. (2006). Online collaboration principles. *Journal of Asynchronous Learning Networks*, 10(1), 25-33.
- Golder, S. A. & Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *Journal of Information Science*, 32(2), 198.
- Gorczyca, A. (2002). Changes in group communication in the context of “virtual/real ratio”. In *20Th ecaade conference proceedings - connecting the real and the virtual - design e-ducation*. Warsaw, Poland.
- Grippa, F., Zilli, A., Laubacher, R., & Gloor, P. A. (2006). E-Mail may not reflect the social network. In *NAACSOS conference*. Notre Dame, IN, USA : North American Association for Computational Social and Organizational Science.
- Gross, M., Yi-Luen Do, E., McCall, R., Citrin, W., Hamill, P., Warmack, A., *et al.* (1998). Collaboration and coordination in architectural design: Approaches to computer mediated team work. *Automation in Construction*, 7(6), 465-473.
- Gruber, T. (2007). Ontology of folksonomy: A mash-up of apples and oranges . *The International Journal on Semantic Web & Information Systems*, 3(2).
- Gulli, A. & Signorini, A. (2005). The indexable web is more than 11.5 billion pages. In *Special interest tracks and posters of the 14th international conference on world wide web*. Chiba, Japan.
- Halpin, H., Robu, V., & Shepard, H. (2006). The dynamics and semantics of collaborative tagging. In *Proceedings of the 1st semantic authoring and annotation workshop (SAAW 06)*. Georgia, USA.
- Harrison, D. (2008, August 21). Autodesk beyond desktop CAD & BIM. *Stressfree solutions blog* [Article]. Retrieved April 10, 2010, from http://www.stress-free.co.nz/autodesk_beyond_desktop_cad_and_bim
- Harrison, D. (2009, February 26). Bimserver and the potential of server-side BIM. *Stressfree solutions blog* [Web page]. Retrieved May 20, 2010, from http://www.stress-free.co.nz/bimserver_and_the_potential_of_serverside_bim
- Harrison, D. (2010, January 7). Bluestreak and the birth of a collaboration kernel. *Stressfree solutions blog* [Web page]. Retrieved June 6, 2010, from http://www.stress-free.co.nz/autodesk_bluestreak_and_the_birth_of_a_collaboration_kernel
- Haymaker, J., Fischer, M., Kunz, J., & Suter, B. (2004). Engineering test cases to motivate the formalization of an AEC project model as a directed acyclic graph of views and

- dependencies. *Itcon Journal*, October, 9, 419-441. Retrieved June 23, 2009, from the Google Scholar database, http://www.itcon.org/data/works/att/2004_30.content.05100.pdf
- Heidrich, F., Russell, P., & Stachelhaus, T. (2007). Intervision3D: Online 3D visualisation and conferencing. In *Proceedings of the 12th international conference on computer aided architectural design research in Asia*. Nanjing, China.
- Hendler, J. & Golbeck, J. (2008). Metcalfe's law, web 2.0, and the semantic web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(1), 14-20. Retrieved June 29, 2009, from <http://www.cs.umd.edu/~golbeck/downloads/Web20-SW-JWS-webVersion.pdf>
- Heylighen, A., Martin, W. M., & Cavallin, H. (2005). Knowledge sharing in the wild: Building stories' attempt to unlock the knowledge capital of architectural practice. In *Proceedings of CIB W096 architectural management, "special meeting" on designing value: New directions in architectural management*. Denmark.
- Heylighen, A., Martin, W. M., & Cavallin, H. (2007). Building stories revisited: Unlocking the knowledge capital of architectural practice. *Architectural Engineering and Design Management*, 3(1), 65-74.
- Hibbs, C., Jewett, S. C., & Sullivan, M. (2009). *The art of lean software development: A practical and incremental approach*. California, USA: O'Reilly Media, Inc. Retrieved January 15, 2009, from the Google Books database, <http://books.google.co.nz/books?id=0VsK9cVZauQC>
- Highsmith, J. (2002). *Agile software development ecosystems*. Boston, USA: Pearson Education.
- Howard, R. & Petersen, E. (2001). Monitoring communications in partnering projects. *Itcon*, 6, 1-16.
- Husin, R. & Rafi, A. (2003). The impact of internet-enabled computer-aided design in the construction industry. *Automation in Construction*, 12(5), 509-513.
- Ibrahim, M. & Krawczyk, R. (2003). The level of knowledge of CAD objects within the building information model. *Connecting >> Crossroads of Digital Discourse, Proceedings of the 2003 Annual Conference of the Association for Computer Aided Design in Architecture*, 172-177.
- Ibrahim, M. (2006). To BIM or not to BIM, this is NOT the question. In *Communicating space(s) 24th ecaade conference proceedings*. Volos, Greece.
- IIA (2007). *Integrated project delivery: A guide* (1 ed.). USA: The American Institute of Architects. Retrieved April 3, 2010, from <http://www.aia.org/ipdg>

- Ilich, M., Becerik, B., & Aultman, B. (2006). Online collaboration: Why aren't we using our tools? *The Construction Zone*, 6(3), 10.
- Ingirige, B. & Sexton, M. (2007). Intranets in large construction organisations: Exploring advancements, capabilities and barriers. *Journal of Information Technology in Construction (Itcon)*, 12, 409-423.
- Integrated Project Delivery. (2009, May). Integrated project delivery. [Web page]. Retrieved July 11, 2009, from http://www.constructech.com/news/articles/article.aspx?article_id=7619.
- Isaacs, E. A. & Tang, J. C. (1994). What video can and cannot do for collaboration: A case study. *Multimedia Systems*, 2(2), 63-73.
- Jackson, T., Dawson, R., & Wilson, D. (2003). Reducing the effect of email interruptions on employees. *International Journal of Information Management*, 23(1), 55-65.
- Jamieson, M., Thorpe, A., & Tyler, A. (1996). Refocusing collaboration technologies in the construction value system. *Cib W78*.
- Java, A., Song, X., Finin, T., & Tseng, B. (2007). Why we twitter: Understanding microblogging usage and communities. In *Proceedings of the 9th webkdd and 1st SNA-KDD 2007 workshop on web mining and social network analysis*. California, USA.
- Kalay, Y. (2001). Enhancing multi-disciplinary collaboration through semantically rich representation. *Automation in Construction*, 10(6), 741-755.
- Kent, D. C. & Becerik-Gerber, B. (2010). Understanding construction industry experience and attitudes toward integrated project delivery. *Journal of Construction Engineering and Management*, 1, 124.
- Kern, C. (2008, April). My word! *The history of the term 'virtual office'* [Web page]. Washington D.C., USA. Retrieved April 13, 2010, from <http://www.chriskern.net/history/myWord.html>
- Kirkpatrick, M. (2009, May 1). The man who made gmail says real-time conversation is what's next. *Readwriteweb* [Web page]. Retrieved July 3, 2010, from http://www.readwriteweb.com/archives/the_man_who_made_gmail_says_real-time_conversation.php
- Kiviniemi, A. (1999). IAI and IFC - state of the art. In *Cib W78*. International Council for Research and Innovation in Building and Construction.
- Kiviniemi, A., Fischer, M., & Bazjanac, V. (2005). Integration of multiple product models: IFC model servers as a potential solution. In *CIB W78*. International Council for Research and Innovation in Building and Construction.

- Kiviniemi, Tarandi, Karlshøj, Bell, & Karud (2008, January 1). *Review of the development and implementation of IFC compatible BIM*. Netherlands: Erabuild. Retrieved June 11, 2009, from <http://www.ebst.dk/file/9498/ReviewoftheDevelopmentandImplementationofIFCcompatibleBIM.pdf>
- Kohavi, R., Henne, R. M., & Sommerfield, D. (2007). Practical guide to controlled experiments on the web: Listen to your customers not to the hippo. In *Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining*. California, USA.
- Kruijff, E. & Donath, D. (2001). Supporting shared architectural understanding. In M. Stellingwerff & J. Verbeke (Eds.), *Accolade-Architecture, collaboration, design*. The Netherlands: Delft University Press (DUP Science).
- Kvan, T. (2000). Collaborative design: What is it? *Automation in Construction*, 9(4), 409-415.
- Laepfle, E., Clayton, M., Johnson, R., & Parshall, S. (2005). Content analysis of web-based collaborative design: Empirical evidence of design process. *2005 Report on University Research by the American Institute of Architects*, 104-117. Retrieved May 3, 2010, from <http://www.aia.org/practicing/research/AIAS077475>
- Laiserin, J. (2002, December 16). *Comparing pommes and naranjas*. The Laiserin Letter. Retrieved April 10, 2006, from <http://www.laiserin.com/features/issue15/feature01.php>
- Lam, W., Chua, A., Williams, J. B., & Lee, C. (2005). Virtual teams: Surviving or thriving? *Australasian Society for Computers in Learning in Tertiary Education (Ascilite)*, 357-360.
- Lamb, E., Reed, D., & Khanzode, A. (2009, September 1). Transcending the BIM hype: How to make sense and dollars from building information modelling. *Aecbytes* [Web page]. Retrieved October 6, 2009, from http://www.aecbytes.com/viewpoint/2009/issue_48.html
- Latham, M. (1994). *Constructing the team: Final report of the government/industry review of procurement and contractual arrangements in the UK construction industry*. London, UK: HMSO.
- Lee, T. B., Hendler, J., Lassila, O., & others (2001). The semantic web. *Scientific American*, 284(5), 34-43.
- Leffingwell, D. & Widrig, D. (2003). *Managing software requirements: A use case approach*. Boston, United States: Pearson Education.
- Levine, R., Locke, C., Searls, D., & Weinberger, D. (2000). The cluetrain manifesto. In *The cluetrain manifesto*. Cambridge, USA: Perseus Publishing.

- Lottaz, C. (2000). Increasing understanding during collaboration through advanced representations. *Itcon*, 5.
- MacLeamy, P. (2010, February 9). Patrick macleamy: BIM, BAM, BOOM! How to build greener, high-performance buildings. *HOK renew* [Web page]. Retrieved March 5, 2010, from <http://hokrenew.com/2010/02/09/bim-bam-boom-how-to-guarantee-greener-high-performance-buildings/>
- MacMillan, S., Steele, J., Kirby, P., Spence, R., & Austin, S. (2002). Mapping the design process during the conceptual phase of building projects. *Engineering Construction and Architectural Management*, 9(3), 174-180. Retrieved May 27, 2009, from the Google Scholar database, <http://www.eclipse-research.co.uk/Academic%20journals/eca253.pdf>
- Maher, M. L., Balachandran, M., & Zhang, D. M. (1995). *Case-Based reasoning in design*. New Jersey, USA: Lawrence Erlbaum. Retrieved January 5, 2010, from the Google Books database, <http://books.google.co.nz/books?id=X0d8uwGKyvoC>
- Makice, K. (2009). *Twitter API: Up and running learn how to build applications with the twitter API* (1 ed.). California, USA: O'Reilly Media, Inc.
- Mark, G., Gudith, D., & Klocke, U. (2008). The cost of interrupted work: More speed and stress. In *Proceeding of the twenty-sixth annual SIGCHI conference on human factors in computing systems*. Florence, Italy.
- Martin, W. M., Heylighen, A., & Cavallin, H. (2003). Building stories. A hermeneutic approach to studying design practice. In *Proceedings of the 5th european academy of design conference*. Barcelona, Spain: EAD.
- Martin, W. M., Heylighen, A., & Cavallin, H. (2005). The right story at the right time. *AI & Society*, 19(1), 34-47.
- Mattessich, P. W. & Monsey, B. R. (1992). *Collaboration: What makes it work. A review of research literature on factors influencing successful collaboration*. St. Paul, Minnesota, USA: Amherst H. Wilder Foundation. (Original work published 1992)
- Matthews, O. & Howell, G. A. (2005). Integrated project delivery: An example of relational contracting. *Lean Construction Journal*, 2(1), 46-61.
- McCall (1999). A web-centric CAD system for collaborative design. In *Proceedings of the eighth international conference on computer aided architectural design futures*. Atlanta, USA.
- McCall, R. & Johnson, E. (1997). Using argumentative agents to catalyze and support collaboration in design. *Automation in Construction*, 6(4), 299-309.
- McKenna, M., Wilczynski, H., & VanderSchee, D. (2006). Capital project execution in the oil and gas industry. USA: *Booz. Allen & Hamilton Inc.*

- Mockapetris, P. V. & Dunlap, K. J. (1995). Development of the domain name system. *ACM SIGCOMM Computer Communication Review*, 25(1), 112-122.
- Moum, A. (2010). Design team stories:: Exploring interdisciplinary use of 3D object models in practice. *Automation in Construction*.
- Nardi, B. A., Schiano, D. J., & Gumbrecht, M. (2004). Blogging as a social activity, or, would you let 900 million people read your diary? *Proceedings of CSCW*.
- O'Brien, W. J., Issa, R. A., Hammer, J., Schmalz, M. S., Geunes, J., & Bai, S. X. (2002). SEEK: Accomplishing enterprise information integration across heterogeneous sources. *ITCon, Special Issue ICT for Knowledge Management in Construction, Vol 7*, 101-124.
- O'Reilly, T. (2005). What is Web 2.0? Design patterns and business models for the next generation of software [Web page]. O'Reilly. Retrieved February 1, 2006, from <http://www.oreillynet.com/lpt/a/6228>
- Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). *The pagerank citation ranking: Bringing order to the web*. California, USA: Technical report, Stanford Digital Library Technologies Project.
- Papamichael, K., Pal, V., Bourassa, N., Loffeld, J., & Capeluto, I. G. (2000). An expandable software model for collaborative decision-making during the whole building life cycle. In *Eternity, infinity and virtuality in architecture, proceedings of the 22nd annual conference of the association for computer-aided design in architecture*. Washington D.C., USA.
- Passant, A., Bojars, U., Breslin, J. G., Hastrup, T., Stankovic, M., & Laublet, P. (2010). An overview of SMOB 2: Open, semantic and distributed microblogging. In *4Th international conference on weblogs and social media, ICWSM*. Washington D.C., USA.
- Passant, A., Hastrup, T., Bojars, U., & Breslin, J. (2008). Microblogging: A semantic and distributed approach. In *Proceedings of the 4th workshop on scripting for the semantic web*. Tenerife, Spain.
- Peng, C., Cerulli, C., Lawson, B., Cooper, G., Rezqui, Y., & Jackson, M. (2000). Recording and managing design decision-making processes through an object-oriented framework. In *Fifth design and decision support systems in architecture and urban planning - part one: Architecture proceedings*. Nijkerk, the Netherlands.
- Postel, J. (1980). Dod standard transmission control protocol. *ACM SIGCOMM Computer Communication Review*, 10(4), 132.
- Prodromou, E., Schwartz, R., & Laporte, L. (2008, August 22). Laconica. *FLOSS weekly episode 37* (Podcast interview with Evan Prodromou, the developer of the Laconica

- micro-blogging system) [Audiovisual Material]. The TWiT Network. Retrieved May 3, 2010, from <http://twit.tv/floss37>
- Pulsifer, D. (2008, October 28). A case for knowledge management in the A/E industry. *Aecbytes* [Web page]. Retrieved May 22, 2009, from http://www.aecbytes.com/viewpoint/2008/issue_41.html
- Raymond, E. (2003, September). *The art of unix programming*. Retrieved March 5, 2007, from <http://www.faqs.org/docs/artu/>
- Recordon, D. & Reed, D. (2006). Openid 2.0: A platform for user-centric identity management. In *Proceedings of the second ACM workshop on digital identity management*. Virginia, USA.
- Reinhardt, W., Ebner, M., Beham, G., & Costa, C. (2009). How people are using twitter during conferences. In *Hornung-Prähauser, V., Luckmann, M.(Hg.): 5Th edumedia conference*. Salzburg, Austria.
- Schneider, S. & Petzold, F. (2009). A virtual design platform: Bridging barriers when designing with computers. In *Computation: The new realm of architectural design. 27Th ecaade conference proceedings*. Istanbul, Turkey.
- Sheppard, S. (2009, November 18). Project bluestreak now available. *It's alive in the lab* [Web page]. Autodesk. Retrieved July 7, 2010, from http://labs.blogs.com/its_alive_in_the_lab/2009/11/project-bluestreak-now-available.html
- Shih, C. (2009). *The Facebook era: Tapping online social networks to build better products*. Boston, USA: Prentice Hall.
- Sinclair, J. & Cardew-Hall, M. (2008). The folksonomy tag cloud: When is it useful? *Journal of Information Science*, 34(1), 15.
- Sire, S., Bogdanov, E., Palmér, M., & Gillet, D. (2009). Towards collaborative portable web spaces. In *4Th european conference on technology enhanced learning (EC-TEL) - workshop on mash-up personal learning environments (MUPPLE'09)*. Nice, France.
- Soubra, S., Coudret, F., Duchon, J., Torguet, P., & Gobbetti, E. (2000). Virtual integrated design and construction. In *W78: Construction information technology 2000 taking the construction industry into the 21st century*. Reykjavik, Iceland.
- Starbird, K. & Stamberger, J. (2010). Tweak the tweet: Leveraging microblogging proliferation with a prescriptive syntax to support citizen reporting. In *Proceedings of the 7th international ISCRAM conference*. Seattle, USA.
- The Construction Users Roundtable (2004, August). *Collaboration, integrated information, and the project lifecycle in building design, construction and operation*. The Construction Users Roundtable.

- Tractinsky, N., Katz, A. S., & Ikar, D. (2000). What is beautiful is usable. *Interacting with Computers*, 13(2), 127-145.
- Underwood, G. M., Maglio, P. P., & Barrett, R. (1998). User-Centered push for timely information delivery. In *Proceedings of the seventh international world wide web conference (WWW7)*. Brisbane, Australia: Elsevier. Retrieved January 10, 2010, from <http://www.almaden.ibm.com/cs/wbi/papers/www7/user-centered-push.html>
- Van Damme, C., Hepp, M., & Siorpaes, K. (2007). Folksonomy: An integrated approach for turning folksonomies into ontologies. *Bridging the Gap Between Semantic Web and Web*, 2, 57-70.
- Varkonyi, V. (2009, March 6). Interdisciplinary collaboration strategies in the age of BIM. *Aecbytes* [Web page]. Retrieved May 22, 2009, from http://www.aecbytes.com/viewpoint/2009/issue_43.html
- Vieweg, S., Hughes, A. L., Starbird, K., & Palen, L. (2010). Microblogging during two natural hazards events: What twitter may contribute to situational awareness. In *Proceedings of the 28th international conference on human factors in computing systems*. Georgia, USA.
- Vollmer, G. & Gaßner, K. (2005). Quality improvement of email communication in work groups and organizations by reflection. *Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work*, 124-127.
- Whitehead, E. J. & Goland, Y. Y. (1999). Webdav: A network protocol for remote collaborative authoring on the web. In *Proceedings of the sixth conference on european conference on computer supported cooperative work*. The Netherlands.
- Whittaker, S. & Sidner, C. (1996). Email overload: Exploring personal information management of email. In *Proceedings of the SIGCHI conference on human factors in computing systems: Common ground*.
- Wilkinson, P. (2005). *Construction collaboration technologies: The extranet evolution*. Oxon, UK: Taylor & Francis.
- Willis, D. & Woodward, T. (2005). Diminishing difficulty - mass customization and the digital production of architecture. *Harvard Design Magazine*, 23, 71-83.
- Woestenenk, K. & (2002). The lexicon: Structuring semantics. *Cib W78*.
- Wong, J. & Hong, J. I. (2007). Making mashups with marmite: Towards end-user programming for the web. In *Proceedings of the SIGCHI conference on human factors in computing systems*. California, USA.

- Wong, K. (2009, March 13). Vuuch: The marriage of social media and product design. *Kenneth wong's virtual desktop* [Web page]. Retrieved July 7, 2010, from http://www.deskeng.com/virtual_desktop/?p=235
- Yarrow, J. (2010, April 14). Twitter finally reveals all its secret stats. *Business insider* [Web page]. Retrieved July 7, 2010, from <http://www.businessinsider.com/twitter-stats-2010-4>
- Young, N. W., Jones, S. A., & Bernstein, H. M. (2007). *Interoperability in the construction industry*. McGraw Hill Construction. SmartMarket Report.
- Zeisel, J. (1984). *Inquiry by design: Tools for environment-behavior research*. Cambridge, UK: Cambridge Univ Press. (Original work published 1981) Retrieved January 29, 2010, from the Google Books database, <http://books.google.co.nz/books?id=AwU4AAAAIAAJ>