Approaches to Interdependency: early design exploration across architectural and engineering domains

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

Paul Nicholas
B.Arch (Hons).

School of Architecture and Design
Design and Social Context Portfolio
RMIT University
March 2008
Declaration

I certify that except where due acknowledgement has been made, the work is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program; and, any editorial work, paid or unpaid, carried out by a third party is acknowledged.

Paul W. Nicholas

31.03.08
Acknowledgements

My experiences of the previous three years and their outcome, this thesis, owe a substantial debt of gratitude to many people at RMIT University and Arup. For the opportunity to conduct this research, and for their continuing support, insight and guidance, I am particularly grateful to my supervisor, Professor Mark Burry, and secondary supervisor Andrew Maher. I am also thankful for Dr. Anitra Nelson's advice on interviewing and social science research instruments, and for the invaluable editing provided by Dr. Juliette Peers.

Arup engineers and the products of their work are often referred to in this thesis, and the direction and development of my inquiry has to a large extent been made possible by their influence. Peter Bowtell has provided me with the opportunity to conduct this research within Arup as well as the freedom to find my own role within that practice. Peter has been a constant source of support, advice and opportunity. Additionally, close collaboration on project work with Arup engineers Dr. Joseph Correnza, Tai Hollingsbee, Jon Morgan and John Bahoric would not have been possible but for their enthusiasm, support and willingness to accommodate these research efforts. I would also like to thank my interview subjects for sharing their thoughts and experiences, and to thank Peter, Joseph and Tristram Carfrae for finding the time to read, comment upon and edit aspects of this work.
ABSTRACT

While 3D digital design tools have extended the reach of architectural and engineering designers within their own domains, an approach to practice whereby the architect designs (synthesises), the engineer solves (analyses) and the fabricator makes – in that order – have limited the opportunities for the two disciplines to collaborate during the early design phase. While it is suggested that 3D digital design tools can facilitate a more integrated approach to early design exploration, this idea remains largely untested in practice.

My research considers the extent to which the 3D digital environment might offer different modes of interaction, and potentially new forms of interdependent working, between architectural and engineering designers. My central proposition is that the 3D digital environment can enable interdependent approaches to design which intersect crucial aspects of architectural and engineering exploration during the early design phase which, before the entry of the computer, were otherwise impossible to affect.

A framework for the enquiry is developed firstly through selected organisation theory and design literature and secondly with information gathered from experts in the field via interview. The proposition has been tested through practice-based projects undertaken during a three year postgraduate internship within the Melbourne Australia office of the engineering consultancy Arup. Exploring this problem within the constraints and possibilities of live practice, a ‘living laboratory’, represents a key contribution of the thesis.

Thesis Supervisor: Mark Burry

Title: Professor of Innovation (Spatial Information Architecture) and Director of Spatial Information Architecture Laboratory (SIAL), RMIT University, Melbourne, Australia
Table Of Contents

List Of Figures .................................................................................................................................................. 8
List Of Interviewees......................................................................................................................................... 13

Chapter One: Introduction ............................................................................................................................... 14
Introduction ....................................................................................................................................................... 15
1.1 Thesis Statement And Themes .................................................................................................................. 16
1.2 Research Motivation .................................................................................................................................. 17
1.3 Architect-Engineer Design Exploration ...................................................................................................... 20
1.4 The Research Context: Arup Melbourne ..................................................................................................... 21
1.5 Research Structure ..................................................................................................................................... 22

Chapter Two: Approach To Research And Method ....................................................................................... 30
Introduction ....................................................................................................................................................... 31
2.1 Embedded Research Within Practice: The Academic Context ................................................................. 32
2.2 Arup Melbourne: The Practice Context ...................................................................................................... 32
  2.2.1 Locating The Practice ............................................................................................................................. 33
  2.2.2 Digital Tools Within The Structures Group ............................................................................................ 34
  2.2.3 Technician/Engineer Interaction ............................................................................................................. 34
  2.2.4 3d Tools .................................................................................................................................................. 35
  2.2.5 Changing Practices ................................................................................................................................. 36
2.3 The Participant-Observer Role .................................................................................................................. 37
  2.3.1 Participant-Observation ......................................................................................................................... 37
  2.3.2 Issues In Participant-Observation ......................................................................................................... 38
  2.3.3 Accessing Practice Knowledge Through Participant-Observation .................................................... 39
2.4 Research Instrument One: Interviews ....................................................................................................... 40
  2.4.1 Purpose ................................................................................................................................................ 40
  2.4.2 Participants .......................................................................................................................................... 40
  2.4.3 Interview Technique ............................................................................................................................. 40
  2.4.4 Question List ....................................................................................................................................... 41
  2.4.5 Analysis .............................................................................................................................................. 41
Chapter Three: Factors for Separation and Integration in Architectural and Engineering Design

Introduction

3.1 Common Origins: The Master Builder

3.2 Forces for Architect-Engineer Separation

3.2.1 Forces for Separation: Education

3.2.2 Forces for Separation: Materials

3.2.3 Forces for Separation: Representation

3.2.4 Forces for Separation: Design Processes

3.2.5 Similarity

3.3 Architects versus Engineers

3.3.1 Role Characterisation

3.3.2 A Personification

3.4 The Current Context for Design Integration

3.4.1 The Current Process

3.4.2 Collaboration and 3D Tools

3.4.3 The Early Design Phase

Summary

Chapter Four: Unravelling Interdependency

4.1 Cooperation, Coordination and Collaboration

4.2 Situating Collaboration

4.3 Factors Significant to Interdependency

4.3.1 First Site of Intersection: Differing Perceptions

4.3.2 Second Site of Intersection: Shared And Creative Problem Solving

4.3.3 Third Site of Intersection: Communication

4.3.4 Fourth Site of Intersection: Trust

4.4 Interdependency In Design

Summary
Chapter Five: Intersections & Interdependency Between Architects And Engineers .....86

Introduction .................................................................................................................................................87
5.1 Differing Perceptions: Problems of Interpretation Between CAD and Analytic Software ........ 88
  5.1.1 Translation and Interpretation.......................................................................................................89
  5.1.2 The Experiences and Understandings of Practice .........................................................................91
  5.1.3 Cad Approaches to Translation and Interpretation ......................................................................94
  5.1.4 A Pragmatic Approach ...............................................................................................................96
Synopsis ......................................................................................................................................................99
5.2 Shared and Creative Problem Solving: Performance-Based Design ..............................................100
  5.2.1 Enabling Performance-Based Analysis and Design .................................................................101
  5.2.2 Early Computational Performance-Based Design .......................................................................102
  5.2.3 Current Barriers to Synthesis .................................................................................................104
Synopsis ..................................................................................................................................................105
  5.2.4 The Experiences and Understandings of Practice ......................................................................105
  5.2.5 Optimisation or Optimally Directed Exploration? ....................................................................109
  5.2.6 Contemporary Approaches to Performance-Based Design ....................................................111
Synopsis ..................................................................................................................................................114
5.3 Communication: Back-Propagating Design Information ..............................................................115
  5.3.1 Back-Propagating Design Information ....................................................................................115
  5.3.2 The Experiences and Understandings of Practice ......................................................................117
  5.3.3 Constraints, Requirements and 3D Digital Design Tools .........................................................119
Synopsis ..................................................................................................................................................122
5.4 Trust: Representation and Explication ..........................................................................................123
  5.4.1 Trust in the Architect Engineer Relationship ...........................................................................124
  5.4.2 The Experiences and Understandings of Practice ......................................................................125
  5.4.3 Trust and Communication .......................................................................................................126
Synopsis ..................................................................................................................................................127
Summary ...................................................................................................................................................128

Chapter Six: Project-Based Testing ...................................................................................................131
Introduction .................................................................................................................................................132
6.1 Design_Arch | Design_Eng .................................. ........................................................................133
  6.1.1 Bendigo Canopy .......................................................................................................................135
  Project Description .............................................................................................................................136
  Reconciling Two Diagrams ...............................................................................................................136
  Defining a ‘Rule Of Thumb’ Based Model .....................................................................................137
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Building Definition</td>
<td>182</td>
</tr>
<tr>
<td>Design File</td>
<td>182</td>
</tr>
<tr>
<td>Radiance Simulation</td>
<td>183</td>
</tr>
<tr>
<td>Integration and Extension</td>
<td>183</td>
</tr>
<tr>
<td>A Subsequent Approach</td>
<td>189</td>
</tr>
<tr>
<td>Project Summary</td>
<td>192</td>
</tr>
<tr>
<td>6.2.3 Skybridge</td>
<td>194</td>
</tr>
<tr>
<td>Project Description</td>
<td>195</td>
</tr>
<tr>
<td>The Model</td>
<td>195</td>
</tr>
<tr>
<td>The Analysis Loop</td>
<td>195</td>
</tr>
<tr>
<td>Project Summary</td>
<td>196</td>
</tr>
<tr>
<td>Synopsis</td>
<td>196</td>
</tr>
<tr>
<td>6.3 Design</td>
<td>201</td>
</tr>
<tr>
<td>6.3.1 The Travellers</td>
<td>202</td>
</tr>
<tr>
<td>Project Description</td>
<td>203</td>
</tr>
<tr>
<td>Early Costing and Material Purchase</td>
<td>204</td>
</tr>
<tr>
<td>Full Scale Prototyping</td>
<td>204</td>
</tr>
<tr>
<td>Geometric Rationalisation</td>
<td>205</td>
</tr>
<tr>
<td>Structural Analysis</td>
<td>207</td>
</tr>
<tr>
<td>Building Without Drawing</td>
<td>208</td>
</tr>
<tr>
<td>Manufacturing Process</td>
<td>209</td>
</tr>
<tr>
<td>Project Summary</td>
<td>210</td>
</tr>
<tr>
<td>6.3.2 Rectangular Pitch Stadium</td>
<td>213</td>
</tr>
<tr>
<td>Project Description</td>
<td>214</td>
</tr>
<tr>
<td>CATIA Model - Ellipsoids</td>
<td>214</td>
</tr>
<tr>
<td>Panel Rationalisation</td>
<td>215</td>
</tr>
<tr>
<td>Project Summary</td>
<td>217</td>
</tr>
<tr>
<td>Synopsis</td>
<td>217</td>
</tr>
<tr>
<td>Summary</td>
<td>220</td>
</tr>
</tbody>
</table>

**Chapter Seven: Discussion And Conclusions** ......................................................... 221

7.1 Conceptual Paradigms ......................................................................................... 223
7.2 Practice-Based Outcomes ................................................................................... 229
Appendix ......................................................................................................................... 231

A1. Interviews ...................................................................................................................... 232
   A1.1 Mr. A Interview Transcript ....................................................................................... 232
   A1.2 Mr. B Interview Transcription ................................................................................. 235
   A1.3 Mr. E Interview Transcription ................................................................................. 238
   A1.4 Mr. G Interview Transcript ....................................................................................... 241

A2. Something Rich and Strange ....................................................................................... 244

A3. Published Papers ........................................................................................................ 246
   A3.1 No Place for Drones ............................................................................................... 247
   A3.2 Import A2: Interpretation and Precision Tools......................................................... 254

A4. Scripts ........................................................................................................................ 232
   A4.1 RhinoScript (for Bendigo Canopy) .......................................................................... 232
   A4.2 Simulated Annealing Script For Ecotect (For Cheese Tower)............................... 235
   A4.3 Excerpt From Catia™ Vba™ Script (For Marina Bay Bridge)................................. 238

A5. Enlarged Figures ......................................................................................................... 274
   A5.1 Figure 86 (Structural model within CATIA, showing loads and restraints)............. 275
   A5.2 Figure 87 (False colour plot of stress analysis - optimised model)............................ 275
   A5.3 Figure 88 (False colour plot of stress analysis - unoptimised model)....................... 276
   A5.4 Figure 96 (Multi-criteria optimisation - false colour plot showing stress)............. 276
   A5.5 Figure 97 (Fixed branching points along the short sides of the canopy)................. 277

Bibliography ......................................................................................................................... 278
List of Figures

Figure 1. KBHwater. Final geometric model ................................................................. 18
Figure 2. Detail from scripted design exploration .......................................................... 18
Figure 3. Geometric results from scripted facade generation ...................................... 18
Figure 4. Serpentine Pavilion (2002). Sequence describing the scripted geometry generation process
    (Image source: Arup, Leach et al 2004) ................................................................. 20
Figure 5. Serpentine Pavilion (2002). Built structure (Image source: Arup, Leach et al 2004) ................................................................. 20
Figure 6. Extension to the Victoria & Albert (1996). Design exploration via adjusting variables within a script
    (Image source: Balmond 2002) ................................................................. 20
Figure 7. My workstation within the Melbourne Australia Arup office ........................................ 37
Figure 8. Proposed stadium in China - option 1 .......................................................... 42
Figure 9. Proposed stadium in China - option 2 .......................................................... 42
Figure 10. Proposed athletics stadium in China - option 1 ............................................ 43
Figure 11. Proposed athletics stadium in China - option 1 ............................................ 43
Figure 12. Proposed athletics stadium in China - option 2 ............................................ 43
Figure 13. Proposed athletics stadium in China - option 2 ............................................ 43
Figure 14. Proposed canopy for Law Courts in Melbourne - option 1 ............................. 44
Figure 16. Proposed canopy for Sports Hub in Singapore ............................................... 44
Figure 18. Proposed canopy in Melbourne - option 1 .................................................. 44
Figure 19. Proposed swimming centre in China ........................................................... 44
Figure 15. Proposed canopy for Law Courts in Melbourne - option 2 ........................... 44
Figure 17. Proposed canopy for Sports Hub in Singapore ............................................... 44
Figure 20. Proposed canopy in Melbourne - option 2 .................................................. 44
Figure 21. Proposed swimming centre in China ........................................................... 44
Figure 22. Proposed facade in Singapore ................................................................. 45
Figure 23. Proposed cultural centre in Singapore - framing diagram .............................. 45
Figure 24. Proposed tensegrity bridge in Singapore ...................................................... 45
Figure 25. The master builder ................................................................................. 49
Figure 26. The split between designers and makers ..................................................... 49
Figure 27. The split between designers ........................................................................ 49
Figure 28. One Shell Plaza, Houston 1968, Skidmore Owings Merrill and Fazlur Kahn
    (Image source: Billington 1983) ......................................................................... 54
Figure 29. Hanging model for the Colonia Guell Church 1889-1908, Antoni Gaudi ................................. 54
Figure 30. Ponte sul Basento 1976, Sergio Musmeci
    (Image source: http://img444.imageshack.us/img444/2568/musmc7.jpg) ................ 54
Figure 31. A comparative study of architects and contractors (Image source: Ankrah & Langford 2005) .................. 55
Figure 33. Three stages of design - generate, analyse & decide (Image source: Flager & Haymaker 2007) ........ 60
Figure 32. Determining a structural solution post architectural design -
    an example from the Melbourne Arup office ....................................................... 60
Figure 35. Existing Design Process (Image source: Pressman 2007, adapted) ................. 62
Figure 36. Preferred Design Process (Image source: Pressman 2007, adapted) ............... 62
Figure 34. The fall-off of design information between each phase of design development
    (Image source: Bernstein 2005, adapted) .......................................................... 62
Figure 37. Cooperation ......................................................................................... 71
Figure 38. Coordination ......................................................................................... 71
Figure 39. Collaboration ......................................................................................... 71
Figure 40. Bargaining Vs Collaborative approaches (Image source: Handy 1976, adapted) .... 73
Figure 41. Four sites of architect engineer intersection ............................................... 75
Figure 138. Diagram showing the progression of information through software and scripting stages
Figure 139. The building envelope is divided into 750X750mm quads
Figure 140. Points, located +500mm above the floor, and zones for testing
Figure 141. Wireframe of geometric model showing floorplates
Figure 142. Diagram showing ‘hits’ on quads from rays cast from a test point
Figure 143. False colour plots showing ‘hits’ recorded within each zone
Figure 144. Specifying different programmatic configurations within the design file
Figure 145. Diagram showing possible configurations
Figure 146. Program sensitivity to direct sunlight
Figure 147. Four examples of differing window configurations generated after specifying different program configurations
Figure 148. Competition submission
Figure 149. A typical shopping centre atrium
Figure 150. The project site and context
Figure 151. Contextual design geometry includable in the model
Figure 152. Geometric model setup within GC
Figure 153. Varying model parameters within GC
Figure 154. Sequence describing an iterative, additive process whereby the definition of new floor quad elements is constrained to the edges of existing floor quad elements
Figure 155. Diagram of the above described sequence
Figure 156. Diagram showing final analysis/synthesis process
Figure 157. Diagram showing iterative opacity process - ‘greyer’ quads are more opaque
Figure 158. False colour plots showing daylight factor measured over 10 iterations
Figure 159. Colour plots showing iterative increase in opacity over 10 iterations
Figure 160. Re-importing analytic data back into the GC model - basic model with column grid
Figure 161. Re-importing analytic data back into the GC model - testing a minimum DF of 5%
Figure 162. Re-importing analytic data back into the GC model - testing a minimum DF of 2%
Figure 163. Using the GC ‘slider’ to test different minimum daylight factors
Figure 164. The analysis grid within Radiance
Figure 165. False colour plot showing second floor DF
Figure 166. False colour plot showing first floor DF
Figure 167. False colour plot showing ground floor DF
Figure 168. Diagram showing shadows cast by the sun at the site over the period of a year
Figure 169. Diagram showing average DF result over 200 iterations of the simulated annealing process
Figure 170. Testing simulation results of different calculation precisions within Ecotect
Figure 171. Geometric results of optimisation process - configuration 1
Figure 172. Geometric results of optimisation process - configuration 2
Figure 173. Typical configuration - average daylight factor throughout building: 2.24649
Figure 174. Optimised configuration - average daylight factor throughout building: 3.24362
Figure 175. Defining loads and restraints within the CAD model
Figure 176. Random point generation within the CAD model
Figure 177. Automating the analytic process from Excel
Figure 178. Iterative form finding process - test run with single central load and four restraints
Figure 179. Iterative form finding process - pedestrian bridge with multiple loads and restraints
Figure 180. Form-found pedestrian bridge - perspective
Figure 181. Form-found pedestrian bridge - elevation
Figure 182. Form-found pedestrian bridge - perspective plan
Figure 183. The project site: Sandridge Bridge, Melbourne Australia
Figure 184. The Travellers sculptures - early sketches
Figure 185. Geometric definition of each sculpture
Figure 186. Engineer, fabricators and project manager in early design discussions .................................................................203
Figure 187. Sealed wax prototypes for 3 possible connection details ..........................................................................................204
Figure 188. 1:1 steel prototypes of chosen connection detail ..................................................................................................204
Figure 189. 1:1 prototype of Technoman’s head .........................................................................................................................204
Figure 190. Connection option 1: preferred engineering option .................................................................................................205
Figure 191. Connection option 2: preferred fabrication option ..................................................................................................205
Figure 192. Connection option 3: negotiated option ....................................................................................................................205
Figure 193. Geometry rationalisation - curves are defined by arc segments ..................................................................................206
Figure 194. Comparison between original spline curves and rationalised curves ........................................................................206
Figure 195. Steps in rationalisation script ..................................................................................................................................206
Figure 196. Structural analysis - resolved translation ..................................................................................................................207
Figure 197. Structural analysis - combined stress .........................................................................................................................207
Figure 198. Steel planar curved members for one Traveller .........................................................................................................208
Figure 199. Steel diagonal members for one Traveller ................................................................................................................208
Figure 200. Consulting a plotted centreline template ..................................................................................................................208
Figure 201. Documentation within Excel spreadsheet ................................................................................................................209
Figure 202. Excel spreadsheet automatically synchronised with model .....................................................................................209
Figure 203. Plotted centre-line template .........................................................................................................................................209
Figure 204. Selecting members within a 3D model from the Excel spreadsheet ............................................................................209
Figure 205. Installation of a sculpture on site ..................................................................................................................................212
Figure 206. The Travellers on site 1 .................................................................................................................................................212
Figure 207. The Travellers on site 2 ..................................................................................................................................................212
Figure 208. The Technoman sculpture - detail ..............................................................................................................................212
Figure 209. The RPS roof geometry .................................................................................................................................................214
Figure 210. Roof geometry defined by sightlines ..........................................................................................................................214
Figure 211. Roof geometry defined by set extents ........................................................................................................................214
Figure 212. Diagram showing discrepancy from the mean in panel surface area ........................................................................214
Figure 213. Diagram showing discrepancy from the mean in the angle of members meeting nodes ........................................215
Figure 214. Histogram showing discrepancy from the mean in the angle of members meeting nodes ........................................215
Figure 215. Investigating ellipsoid-based geometry for the shells within CATIA ............................................................................215
Figure 216. Varying the parameters defining the ellipsoid ........................................................................................................215
Figure 217. Segmenting the ellipsoid-based shell into panels ........................................................................................................216
Figure 218. Comparison of ellipsoid-based roof against ellipse-based roof geometry ................................................................216
Figure 219. Analysis of panel shapes from one roof bay ..................................................................................................................216
Figure 220. Bay 1 - panels coloured according to surface area ...................................................................................................218
Figure 221. Bay 2 - panels coloured according to surface area ...................................................................................................218
Figure 222. Bay 3 - panels coloured according to surface area ...................................................................................................218
Figure 223. Bay 4 - panels coloured according to surface area ...................................................................................................218
Figure 224. Bay 5 - panels coloured according to surface area ...................................................................................................218
Figure 225. Bay 6 - panels coloured according to surface area ...................................................................................................218
Figure 226. Identifying panel groupings that could be rationalised ..............................................................................................219
Figure 227. Panel groupings shown on shells ..............................................................................................................................219
## List of Interviewees

<table>
<thead>
<tr>
<th>Name</th>
<th>Date</th>
<th>Office</th>
<th>Position</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mr. A</td>
<td>May-06</td>
<td>Melbourne</td>
<td>Associate</td>
<td>A structural engineer with experience in large scale airport and stadia projects in the UK and USA</td>
</tr>
<tr>
<td>Mr. B</td>
<td>May-06</td>
<td>Melbourne</td>
<td>Senior Engineer</td>
<td>A sustainability consultant and building services engineer. He provides specialist advice on building performance and building codes</td>
</tr>
<tr>
<td>Mr. C</td>
<td>May-06</td>
<td>Melbourne</td>
<td>Senior Engineer</td>
<td>A specialist in Computational Fluid Dynamics (CFD) and environmental physics issues including internal and external microclimate, and internal and external daylight and solar access</td>
</tr>
<tr>
<td>Mr. D</td>
<td>Nov-06</td>
<td>London</td>
<td>Associate</td>
<td>An Arup researcher with an architectural background and an overview role within Arup globally</td>
</tr>
<tr>
<td>Mr. E</td>
<td>Nov-06</td>
<td>London</td>
<td>Associate Director</td>
<td>An architect with particular experience in complex geometries and scripting</td>
</tr>
<tr>
<td>Mr. F</td>
<td>Nov-06</td>
<td>London</td>
<td>Associate</td>
<td>A structural engineer with experience in complex geometries, computer integrated structural design and digital fabrication</td>
</tr>
<tr>
<td>Mr. G</td>
<td>Nov-06</td>
<td>London</td>
<td>Director</td>
<td>An architect. A specialist in sports venue design and sports architecture.</td>
</tr>
<tr>
<td>Mr. H</td>
<td>Dec-06</td>
<td>Hong Kong</td>
<td>Senior Engineer</td>
<td>A structural engineer with extensive experience working on One Island East, Hong Kong</td>
</tr>
<tr>
<td>Mr. I</td>
<td>Dec-06</td>
<td>Hong Kong</td>
<td>Associate</td>
<td>A structural engineer with extensive experience working on CCTV, Beijing</td>
</tr>
</tbody>
</table>
Chapter One: Introduction

1.1 THESIS STATEMENT AND AIMS
1.2 RESEARCH MOTIVATION
1.3 ARCHITECT-ENGINEER DESIGN EXPLORATION
1.4 THE RESEARCH CONTEXT: ARUP MELBOURNE
1.5 RESEARCH STRUCTURE
1.6 INDEX OF TERMS
INTRODUCTION

My research considers the extent to which the 3D digital environment might offer different modes of interaction, and potentially new forms of interdependent working, between architectural and engineering designers. My central proposition is that 3D digital tools and representations can enable interdependent approaches to design which intersect aspects of architectural and engineering exploration during the early design phase. This involves working across disciplinary boundaries to more effectively guide and inform early design exploration, potentially leading to better designs or better design processes.

My thesis has developed around a series of eight practice-based projects that I have undertaken within the Melbourne office of the engineering consultancy Arup, which explore the linkages between computational analysis, making and early design exploration. These projects demonstrate the use of the 3D digital environment to step outside an approach to practice whereby the architect designs, the engineer solves and the fabricator makes – in that order – and provide evidence for the above claim. Exploring this problem within the constraints and possibilities of live practice, a ‘living laboratory’, represents a key contribution of the thesis.
1.1 THESIS STATEMENT AND THEMES

My research has developed around and explored four themes – differing perceptions, shared and creative problem solving, communication and trust – each of which pose problems for interdependent working between architects and engineers. My practice-based project work, the primary vehicle for testing, explores these themes against issues of design generation and of design realisation in engineering and architectural practice. I have developed a conceptual framework which enables this investigation using organisation theory, design theory and the experiences and understandings of nine senior Arup practitioners captured by interview.

There is currently a great interest in the potential for 3D digital tools and methods to facilitate working across what have been the traditional boundaries of architectural and engineering practice. An example of the current state of the discourse occurs in the forward to the 2007 ‘ScriptedByPurpose’ exhibition, where Leach states that:

“...the real potential of scripting lies perhaps beyond questions of innovative form. For what this realm offers is not so much an extension to postmodern scenographic form-making, but a critique of that realm. With increasing concerns for sustainability and efficiency, the need to optimize performance in terms of environmental, structural, economic and other concerns, demarcates a new ethical horizon of possibilities. Digital tools can be used not only to model and test performance, but also to generate buildings in the first place. In other words, once a performative logic has been written into a script, the results are already optimized”
Leach, 2007

Similarly, Ceccato has observed that “a result-driven paradigm is [being] replaced by a process-driven paradigm, in which results are the inevitable outcome of the process, but where the true power lies not in the product but in the system that creates it. By changing, guiding or optimising the process, the product can be consistently improved, diversified or focused as required by specific circumstances. This emergent paradigm is becoming reality through the application of computing technologies and methodologies” (Ceccato 2001: n.p.). The question neither author raises, and which I found myself stumbling into prior to commencing this research, is by what mechanisms should architects engage with these different ‘performative logics’ (drawn either from analysis or fabrication) as Leach terms them which, at least in an objective, measurable form, have not been part of the early design phase and arguably the modern architectural domain?

As the answer to this question seems to be unresolved within both the research and practice communities, my research will help inform this emerging discussion, where several different approaches are currently in play. The first of these is simplification: the sun, for example, can be geometrically interpreted as a fixed point in space, towards which openings can be oriented and sized accordingly – a number of scripts for different CAD software have been written to automate this process, and in fact it is what architects do whenever they produce a shadow study.

1 ScriptedByPurpose, in which the author exhibited work, was held at the FUEL Collection gallery in Philadelphia, USA, in September 2007. The exhibition was curated by Marc Fornes and Skylar Tibbits.
The results of such a process might be later simulated and analysed by engineers, who would then suggest alterations to the design. This approach can be thought of as a sequentially dependant process.

Secondly, architects might learn the analytical tools normally used by engineers and apply them within the design process as tools of synthesis: the use of conceptually-geared analysis and simulation software by architectural offices is a possible model. However, this approach demands more than simply using a new software tool – architects would also need to possess the knowledge needed to interpret the results, which would require changes to the current education system. This approach can be thought of as an independent process.

The third possible approach is for architects to work closely with the discipline that does possess this knowledge – engineering - and to make more informed use of this relationship in guiding the design process. While seemingly the most obvious approach, this option poses significant challenges to traditional modes of architect engineer interaction and the role of each party within the design process. This approach can be thought of as a lateral, interdependent process. Within the concept of interdependence however, there is an unsettling word: dependence.

Within this thesis, interdependence is defined as follows:

Interdependency is a productive form of practice enabled by mutual and lateral dependence. Interdependent parties use problem solving processes that meet not only their own respective goals, but also those of others, by constructively engaging difference across their boundaries to search for solutions that go beyond the limits of singular domains.

My central proposition is that 3D digital tools can enable interdependencies between crucial aspects of architectural and engineering design exploration during the early design phase, leading to more efficient and more affective design processes and outcomes. This involves working across disciplinary boundaries in ways consistent with 3D digital practice to make information that is typically developed downstream and only acted upon reactively available to help actively guide early design exploration.

1.2 RESEARCH MOTIVATION
I commenced my doctoral research in March 2005, in response to a problem that factored on my experience and inexperience and which I would later find to be the subject of significant research. One part of this problem was a longstanding interest in the architectural use of rule-based and generative design techniques for early design exploration that raised design questions about form, structure and environmental performance that I, as an architect, was not able to answer within my own knowledge domain. Resolution of these issues suggested the active involvement of engineers. The other part of the problem lay in practice-based work that I was

2 Also see the Index of Terms in Chapter One Section 1.6
doing in parallel. Within these practices, some of which pursued complicated geometrical forms, interaction with engineers during the early design phase was notable only for its absence. While trying to reconcile these two design experiences, a question developed: could architects and engineers engage with one another in ways consistent with 3D digital practice?

Prior to commencing my PhD I had developed a practice-led interest in scripting and parametric modeling. Underlying this exploration was an interest in using the computer within design to do things that I could not achieve by other methods. Within these projects, design exploration that made use of generative and parametric methods typically took place at two sites: firstly, in spatial allocation and programming, which I explored through CA (Cellular Automata) and CA-like methods and secondly, in façade articulation, which used the results of the spatial allocation investigations as input. These enquiries were traditionally architectural in their proposition, explored through computational means, and increasingly they began to raise a similar problem. A typical example of this work is the 2004 ‘KBHwater’ project, which investigated the renovation of an existing water silo for residential living in Copenhagen, where the small diameter of the silo required that a new structural façade be added (Fig.1).

The façade design explored a honeycomb geometry that was generated through a scripting process. The script linked several inputs, some of which were accessed directly from the 3D model and some of which were user inputs. These inputs were linked directly to the generation of the honeycomb. The most significant of these were the depth of each cell, responding to the curvature of an underlying surface. Each option has an equal percentage of closed and open cells.
which responded to the curvature of an underlying surface geometry (Fig.2), and user inputs which controlled the percentage of closed or open cells, but not the distribution of those cells (Fig.3), and the scale of the honeycomb. By manipulating the underlying surface geometry within the 3D model, or changing the user inputs, I was able to produce a potentially infinite range of different design variations.

The problem, which is a consequence of being able to explore many variations within a design, can be stated quite simply: in what way is any given variation better than any other? While part of this question can be answered within the architectural realm on compositional grounds and informed personal preference, a potentially significant part of the answer (in my view) lay outside that domain, in engineering: what impact did the different parameters and the subsequent deformation have on aspects of the building's performance, for example the structural capacity of the façade, the cost implications, or the quality and quantity of light entering the apartments? Could these criteria help guide or lead the design exploration? For this to occur, engineering input would have to be an active driver within the generative design process, which would need to engage with and synthesise this input. I thought that, setting aside the scripted element of this work, this sounded like a reasonable account of how architects and engineers interacted within practice. Surely, if I was running into these issues on speculative projects, those who were actually building would be encountering and resolving these same problems.

Parallel to these projects I worked for several Melbourne-based architectural practices. Working in practice, perhaps unsurprisingly, did not offer me a productive means to think further about these issues. There were a number of reasons. In the majority of these practices Computer Aided Design (CAD) was used as a 2D drawing and documentation tool, not as an investigative medium, design information was communicated to other involved parties via 2D drawings and, more significantly, there was little to no interaction with ‘the engineers’ during the design process. There were, however, some exceptions: two practices for whom I worked designed in 3D using digital design techniques drawn from animation and generative tools. But even in these offices, which were pursuing complicated geometrical forms, interaction with engineers was notable only for its absence until relatively late in the process. Given that these architectural practices viewed themselves as ‘design practices’, engineering concerns perhaps were not issues deemed significant enough to impact upon architectural design thinking.

Within these practices, my engagement with engineers was similar to someone whose engagement with architecture occurs entirely through WallpaperTM magazine – exposed to the products but in relative ignorance of the processes and thinking that lay behind them. While a limited context for further investigation, this experience did raise three further questions:

- What factors make early design interaction between architects and engineers uncommon?
- Do these underlying factors stem from social or technical circumstances?
- What might challenge these existing modes of non-engagement?
1.3 ARCHITECT-ENGINEER DESIGN EXPLORATION

While engaged in speculative and practice work I became aware of contemporary built and speculative design work in which architects and engineers were working together throughout the design process. These structures, examples being the 2002 Serpentine Pavilion by Toyo Ito & Associates and Arup (Figs.4-5) and the proposed 1996 Victoria and Albert extension by Studio Daniel Libeskind and Arup (Fig.6), were architecturally engaging, geometrically complicated and designed using 3D modeling software. Particularly interesting to me was that the design exploration for these structures included similar scripted, rule-based techniques to those which I was exploring, and that these techniques formed a mechanism by which both disciplines collaborated and reconciled their differing design concerns. Scripting, in particular, seemed to provide opportunities that traditional modes of representation could not.

Within these projects, issues typically addressed only later in the design process, such as fabrication or structural performance, were connected to the generative process of early design exploration. While these structures showed that early collaborative interaction between architectural and engineering design processes was possible, and arguably resulted in outcomes that neither discipline could have achieved alone, there were a very limited number of examples. These typically involved high profile projects, renowned architects and often an equally famous engineer, large offices, and substantial budgets. While answering some questions, this work raised others:
What potential forms of interaction might alternative forms of information transfer, such as scripting, offer architects and engineers?

What exactly does collaboration entail?

What are the limits to interaction of this kind?

Some of the questions raised are addressed in literature which I reference in the subsequent chapters.

Despite the two disciplines sharing a common origin, the design intentions of architects and engineers may often be conflicting rather than converging. The roles assumed by each party, and consequently their interaction, are varied and not static: the engineer can be a technician, artist or collaborator (Nordenson 2000: 37), while the architect might ignore, accept, symbolize or celebrate engineering concerns (MacDonald 2001: 114). Pressures of time, complication and budget are seen to demand more integrated working processes (Westbury, in Castle 2002: 68), to the extent that “the engineer is now required not so much to calculate the inner skeleton of a design (as was usually the case at the beginning of the century) but to evolve with the architect the very character of the composition itself” (Collins 1960: 31). An increasing number of authors state that to meet these challenges architects and engineers need to collaborate from an early stage (Akin et al 1998, Barrow 2004, Drogemuller et al 2004, Gero 1998, Flemming & Mahdavi 1993, Kennon 2006). Collaboration, as opposed to the closely related processes of coordination and cooperation, is singularly characterised by the concept of interdependence (Gray 1989: 11).

Trying to integrate architectural and engineering design exploration within the early design phase requires that 3D digital tools facilitate the iterative, exploratory nature of that period and support intersection in a context of incomplete and imprecise data (Akin et al 1998: n.p.). But the digital domain is not seamless – multiple issues of translation limit the easy transfer of digital information between the software used by either party (eg. Amor & Faraj 2001: 62, Eastman 1991: 17, Shelden 2006: 82) – however a use of 3D tools that has been mired in the past (Kvan et al 2003: n.p.) has meant that their potential for facilitating integrative strategies has not been fully explored.

1.4 THE RESEARCH CONTEXT: ARUP MELBOURNE

My research has provided an opportunity to improve and share my understanding of the problems involved with architect-engineer interaction. To facilitate this, as a researcher I have had to assume a rather precarious location. Not only have I conducted this research at the intersection of architecture and engineering, but also at the intersections of research and practice and of analysis and synthesis. It has been undertaken within the Spatial Information Architecture Laboratory’s (SIAL) ‘Embedded Practice within Architectural Research’ program, through which I have been ‘embedded’ within a professional design practice, the Melbourne Australia office of the engineering consultancy Arup3. I entered this program specifically to conduct the research

---

3 Further information about Arup and Arup Melbourne can be found in Chapter Two, Section 2.2.1
within this practice, and have consequently had the novel experience of being the only architect working within the Melbourne office, involved in design projects as an active participant, but with adequate time for reflection, discussion, provocation and evaluation. This has provided me with an ‘on-the-ground’ means to investigate the relationship between architects and engineers and the possibilities for digital interaction through live and research projects within Arup. As a result, the projects presented within this thesis represent research that has been road tested in actual practice, offering a unique combination of speculative analyses within real-time issues and conditions.

Arup is a particularly appropriate practice in which to conduct this design-focussed research – it has a long history of working closely with architects on projects that are highly challenging and unconventional, an underlying philosophy articulated by Ove Arup as ‘total architecture’ (Arup 1985: 34), which recognises that the interactive collaboration between different design disciplines leads to better design solutions, and a long experience of the difficulties involved in putting that philosophy into practice. Influential Arup engineers, including Ove Arup and Peter Rice, have written insightfully on these issues (e.g. Arup 1985 & Rice 1998). Indeed, Arup’s Australian involvement began with just such a project, the Sydney Opera House, which has become known for its resolution of complicated geometry, architect-engineer tensions (unresolved) and for pioneering the use of computation in building design (Rice 1998). The practice is deeply interested in the benefits that the 3D digital environment can bring to its workflow, and as a multidisciplinary practice operates as a microcosm of the wider industry, experiencing the same problems within practice as occur between practices. Since 1964 Arup has supported a dedicated research group, Arup Research + Development (AR+D), and opportunity and support for research and development activities occurs at all levels throughout the firm (Arup website AR+D 2008).

1.5 RESEARCH STRUCTURE
To test my central research proposition, that the 3D digital environment can enable alternative design approaches that intersect aspects of architectural and engineering design exploration during the early design phase by increasing interdependency, this thesis uses a process of triangulation that engages firstly with the available literature, secondly with information gathered from experts in the field principally via interview, and lastly testing through practice-based (as opposed to university-based) project work. I examine the current nature of architect-engineer interaction through exploring literature on the theme, and develop a conceptual framework around the notion of interdependency through which to understand the challenges posed by an earlier and more integrated relationship. I then test, through practice-based projects, the use of 3D digital tools and processes to enable alternative approaches that intersect aspects of architectural and engineering design exploration, and critically analyse the outcomes through reference to the framework.
The placement, development and testing of these themes within the subsequent chapters is organised as follows:

Chapter Two, ‘Approach to Research and Method’, details the constraints and possibilities of the ‘Embedded Research within Architectural Practice’ context, within which this work has been undertaken, and describes the Melbourne Australia office of Arup, the practice with whom I have been embedded. These contexts have led to the selection of a particular set of ethnographic research instruments, being the use of semi-structured interviews and the undertaking of practice-based studies as a participant-observer. These modes of testing are explained, and the constraints, limitations and requirements that come with them described.

Within Chapter Three, ‘Factors for Separation and Integration in Architectural and Engineering Design’, I examine selected design literature to detail several factors impacting upon the historic and contemporary relationship between architects and engineers, and to introduce the problem towards which this thesis is addressed. I describe a process of specialisation that has led architects and engineers to see different aspects of a common problem, detail the historical factors for separation, the current relationship between domains and the emerging idea of increased integration during the early design phase. The aim of this section is primarily contextual - to introduce the characters and to understand why their interaction can be difficult - and investigation occurs through the concepts of specialisation and disciplinary roles.

Chapter Four, ‘Unravelling Interdependency’, develops my concept of interdependency. I review selected writing on organisation theory to initialise this concept, which has been defined above. From this and design literature, I identify four sites of intersection significant to an understanding of interdependency within a design context; these are differing perceptions, shared and creative problem solving, communication and trust. These themes, which correlate with my practice experience at Arup Melbourne, are developed to introduce the concepts and vocabulary underlying my research. While initialised from the perspective of organisation theory, these sites of intersection can be productively brought back to inform the architect-engineer relationship.

Chapter Five, ‘Intersections & Interdependency between Architects and Engineers’, grounds these four sites of intersection within contemporary issues of digital architectural and engineering practice. Each site is developed firstly through reference to design literature and secondly through the experiences and understandings of nine senior Arup practitioners as captured through my interviews. The views and experiences of these practitioners are used to locate digital limits to, and potential approaches towards, interdependent design exploration between architects and engineers as they are experienced within and by practice. Through this combination of design literature and grounded experience, I extend:
- the understanding of differing perceptions through reference to problems associated with digital information transfer.
- the understanding of joint and creative problem solving by connecting it to the notion of performance-based design.
- the understanding of communication by focussing it upon the idea of back propagating design information.
- the understanding of trust by connecting it to the management and reduction of perceived complexity and risk.

Chapter Six, ‘Project-based Testing’, details the eight project studies that I have undertaken. These studies are grouped into three discourses

- Design\(\text{(Arch)}\) | Design\(\text{(Eng)}\)
- Design | Analysis
- Design | Making

Each group details one major project, followed by several minor projects in support. The section ‘Design\(\text{(Arch)}\) | Design\(\text{(Eng)}\)’ reports three projects that use a common language of geometry to link architectural and engineering design ideas through geometrical interpretation. The section ‘Design | Analysis’ reports three projects in which analytical tools have been used generatively to actively guide and synthesise design exploration. These projects include both optimisation and form finding processes. The final section, ‘Design | Making’, reports two projects in which 3D digital tools have supported the procurement of detailed fabrication information, around which architectural and engineering design thinking can intersect.

Conclusions are then drawn and discussed in Chapter Seven. In evaluating the research I summarise how 3D digital tools have enabled alternative approaches to differing perceptions, joint and creative problem solving, and increased communication and trust have enabled interdependent architect engineer working. I then draw together the impacts of intersecting 3D, digital aspects of architectural and engineering design exploration during the early design phase, and identify those aspects that require further analysis and research to better enable interactions of this kind.

---

4 I have used the ‘bar’ character to separate the two terms in each relationship. There are two ways of understanding this symbol. As the Sheffer stroke, also called the alternative denial, this symbol denotes a logical operation in which at least one of the operands is false. As a concurrency operator, this symbol is used to indicate processes that execute in parallel. The second definition is taken here.
1.6 INDEX OF TERMS

Applications and Software
The tools used most extensively in this dissertation are Rhino™, CATIA™, and the analytical tools GSA™ and Radiance™. Rhino™, first released in 1998 by McNeel, is a free-form NURBS (Non Rational B-Spline) modeling program. CATIA™ (Conception Assistée Tridimensionnelle Interactive) was first released by Avions Marcel Dassault (AMD, now Dassault Systemes) in 1977. CATIA™ is a parametric modeling tool initially developed for the aviation industry. Radiance™ is a physically based rendering package which blends deterministic and stochastic ray-tracing techniques (Ward 1994: 459), which is commonly used to simulate day and artificial lighting. GSA™ (General Structural Analysis) is a structural analysis package developed by Oasys and Arup.

Communication
In interdependent processes, efficient communication needs to occur at multiple ‘levels’ to provide information supplementary to the minimum. This dissertation examines a specific aspect of communication, the way in which digital representations can facilitate the back-propagation of design information. Within this context, supplementary information includes insight into design goals of other parties, the reasoning behind them and the intentions of the designer.

Constraints, Variables and Parameters
Constraints, variables and parameters are a means by which design information can be encoded, communicated and instrumentalised within CAD software. A constraint is a condition that must be satisfied. A variable is an input, the value of which might change over the course of design exploration. A parameter is a context-dependant quantity (Burry et al 2001: 77) that defines a system and determines the extent of its performance. For example, when designing a rectangular room, a variable might control the length of the room, a parameter might relate the width to the length, and a constraint might prevent the size of the room from exceeding twelve square meters. Within parametric software, constraints, variables and parameters are generally defined graphically. When scripting, the designer defines these same entities in a text-based manner.
Cleaning

The term ‘cleaning’ describes the manual rework that is typically required before 3D geometrical information from CAD software can be used effectively within analytical software. Cleaning is conducted by engineers and CAD technicians, and can involve tasks such as manually tracing over all the surfaces in a 3D model, finding every single intersection point, extracting the centrelines from 3D entities, simplifying that geometry or simply assigning different layers or other information to geometry. Often, cleaning a model can take longer than the subsequent processes of analysis.

Design Tools

The architect and engineer’s traditional design tools have included paper, pencil, scale and slide rule, calculator, tables and small-scale models. These tools are used to explore and represent ideas at multiple resolutions, and to exchange this information with others that are part of the design and building teams. While these tools continue to prove extremely useful within the design process, they are limited by their specific characteristics, similar to any design tool.

The development and use of computational design tools, known as Computer Aided Design (CAD) software, has been partly based on the above methods (Fraser 1995: 66). One significant advantage of these tools over the manual version is that they aid the tedious and error prone process of coordinating many related drawings. A second advantage is that they allow the designer to represent in 3D and with a very high level of precision, designs which would be difficult to understand and consequently communicate by hand.

Differing Perceptions

Differing perceptions occur that can limit interaction across boundaries, however without differences between the parties, the range of possible exchanges would be nonexistent. Different disciplines bring different goals, interpretations, levels of expertise and access to information to a common problem. When the information they share about that problem is too limited, or comes from many independent sources, additional differences of perception can arise.
Early Design Phase

This includes the conceptual and schematic design stages, and continues until detailed design. This phase is characterized by competing requirements, ill-structured problems and the ongoing formulation of geometry, materiality and other design information. One main criteria for exploration during the early design phase is speed, and the purpose of this phase is to obtain better or more accurate information without undertaking detailed designs or documentation. A second is ‘correctness’ – decisions need to be sufficiently informed so as to lead to viable and deliverable design solutions. Within this thesis, I focus upon the early design phase because many of the decisions made during this period (as much as 80% of those made on the project (Drogemuller et al. 2004: n.p.)) are critical to the later outcome. Better informing this period of design, when the impact is greatest and the cost is least, through closer architect engineer interaction leads to more effective and more affective solutions.

Generative design

Generative design is a proposed method for rule-based design. The key feature of a generative design process is that, from the application of a series of basic rules for variation to an initial state, new and perhaps unpredictable information is produced. Generative design processes typically consist of a design representation, a generation mechanism (commonly either grammar-based or evolutionary), and a means for evaluation and acceptance of the new generation.

Performance-based design

Performance-based design is a proposed method. The underlying idea of performance-based design is to “engage in a design practice based on feedback loops between making design decisions and evaluating their environmental impact, as a way to inform the on-going process of design” (Caldas & Norford 2002: 173). In this thesis I understand performance-based design as a generative method, whereby an iterative generate and test process repeats, guiding design development, until a condition is met. Performance-based design is made possible by computer-based 3D modeling and analysis tools, which enable architects and engineers to virtually simulate building performances and to converge the traditionally separate explorations of the qualitative and the quantitative.
Interdependency

The concept of interdependency is central to this thesis, and describes a relationship through which parties whom see different aspects of a common problem can constructively explore their differences and search for solutions that go beyond their own limited vision of what is possible. Within this thesis I define interdependency as a means for matching differences across boundaries which is characterised by mutual dependence, mutual responsibility and shared and creative methods and tasks for problem solving.

Joint and creative problem solving

Joint and creative problem solving occurs through negotiation and evaluation, which requires close coupled design processes and new tools and approaches that are suitable to the early design phase. Negotiation is not viewed as a bargaining tactic, but rather as a creative exercise. The results of joint and creative problem solving neither belongs to one discipline nor completely join different disciplines.

Sequential Approach

The sequential approach is a model for design interaction. Within this thesis, it is understood as follows:

The architect produces a design using either 2D or 3D tools and representations. This information is reduced to 2D files or paper-based drawings and passed to the engineer to overlay a structural system, services and other specialist information. The engineer generates 3D models for simulation within analytic software, and the results of that analysis may verify or require changes to the architectural design. The majority of the time changes are required. These are communicated back to the architect via 2D drawings, PowerPoint slides, text-based email and screen captures. This cyclical process occurs several times before a synthesis is achieved. Design documents, generally 2D drawings, are then sent to a fabricator and potentially form the basis for the fabricators 3D model.

Scripting

Scripting is a text-based method for using design tools. Prior to the development of graphical user interfaces (GUI), working with CAD software was a text based process that involved typing the required commands via a keyboard into the program. Whilst CAD software is now graphically-based, the icons and menus are simply shortcuts
to text based commands. Scripting allows the designer direct access to an application’s commands (‘methods’), as well as to general control structures such as loops, logical and mathematical operators and conditionals which dictate a sequential progression through the methods when the script executes. When scripting, the designer is interacting with the internal workings of the computer, albeit on a very low level. While each CAD program has an endemic scripting language, tools such as Visual Basic for Applications™ (VBA) can be used to control many of these programs externally via automation objects.

Trust

Trust is narrowly defined within this thesis a way of facilitating interdependency by managing apparent risk and perceived complexity - without trust, information is screened and the goals of other perspectives are ignored. This thesis explores how alternative digital representations can facilitate trust by increasing understanding and thereby mitigating perceived risk.

---

6 VBA is an automation controller, meaning that it can access and manipulate automation objects (methods) made available by other applications.
Chapter Two: Approach to Research and Method

2.1 EMBEDDED RESEARCH WITHIN PRACTICE: THE ACADEMIC CONTEXT
2.2 ARUP MELBOURNE: THE PRACTICE CONTEXT
2.3 THE PARTICIPANT-OBSERVER ROLE
2.4 RESEARCH INSTRUMENT ONE: INTERVIEWS
2.5 RESEARCH INSTRUMENT TWO: PRACTICE-BASED PROJECTS
INTRODUCTION
Within the preceding chapter I described my motivations for undertaking this research, introduced the academic and practical contexts that frame it and outlined my central research proposition. In this chapter I establish the research design through which I will test this proposition. In Section 2.1 I briefly describe the academic context of the ‘Embedded Research within Architectural Practice’, the program within which this research has been undertaken, and outline its general aims and intentions.

I introduce the practice context of Arup Melbourne, the engineering consultancy within which I have conducted this research, in Section 2.2. In this section I examine the local nature of this practice, the makeup of the structures group, with whom I have worked most closely, and the status of 3D digital tools within the group. Section 2.3 explains the role that I have played within Arup Melbourne, that of participant-observer.

Sections 2.4 and 2.5 describe the two modes of investigation that have been undertaken to test my research proposition. I have firstly undertaken a process of semi-structured interviewing, in which I have drawn from the experiences of Melbourne and internationally located Arup employees. Secondly, I have actively participated in practice-based projects within the Arup Melbourne office.
2.1 EMBEDDED RESEARCH WITHIN PRACTICE: THE ACADEMIC CONTEXT

This research has been conducted within the framework of the Embedded Research within Architectural Practice program at the Spatial Information Architecture Laboratory (SIAL), Royal Melbourne University of Technology (RMIT), which commenced 1 March 2005. The program as a whole comprises four researchers, of which I have been one, ‘embedded’ within four Melbourne and Sydney-based practices ranging from the small to the large scale1. The program is particular in several respects, most significantly in that it attempts to bridge the research gap between academia and practice. The intention of this section is not to describe this program in detail, but rather to briefly outline its stated aims. These are:

- To investigate four different routes to design practice innovation in four different and unique practice contexts through project-based research.
- To create a better understanding of the factors that lead to change and innovation in design practice.
- To initiate a forum composed of key members of each of the participating practices for dialogue leading to new areas of research and development that will help maintain the competitive position of Australian architectural and engineering design and its role in the construction industry in the world market.

(Embedded Research website, 2005)

As described on the Embedded Research website (2005), the premise of the program is that although “research is an integral component of design practice… the scope for an in depth or generic approach to investigating research questions in Practice is generally limited by the day to day pressures of production schedules”. When conducted, research is often not captured and retained within individual firms but lost through lack of documentation or recognition, and indeed “research in architecture is often seen as unrelated to the practice of architecture”. Practice-based applied research provides an opportunity for practice and the university to “address a wide range of different issues in design collectively that will mutually benefit the participating practices, students, and our community”. Through this approach, both practices and students have the opportunity to “explore how their processes can be mapped onto new digitally-supported and supportive ways of working”. This has the potential to “advance knowledge of better, more sophisticated approaches to design exploration and execution in practice”.

2.2 ARUP MELBOURNE: THE PRACTICE CONTEXT

This section aims to provide an insight into the Arup Melbourne office. Within it, I introduce the office context, outline the types of project undertaken by the office, list the software used within the office and describe the difference between two of its ‘occupants’, the structural engineer and the CAD technician. I lastly list some of the ideas that are driving Arup Melbourne to a much

1 Each of the four researchers within the Embedded Research within Architectural Practice program have conducted separate research and, while there is an obvious future opportunity to synthesise aspects of this research as a whole, within this dissertation it will not be referred to.
increased use of 3D digital tools.

2.2.1 LOCATING THE PRACTICE
Arup is a multinational, multidisciplinary consultancy that provides engineering, architectural and planning services. It employs over 9000 employees throughout 82 offices in 34 countries around the world. The first Arup office to open in Australia was in 1963, to undertake the structural design of the Sydney Opera House, and there are currently seven Arup offices within Australia.

Arup Melbourne occupies the top three floors of ICI House, a location that places the practice firmly at the intersection of Melbourne’s architectural, engineering and artistic history. ICI House, designed by Bates Smart and McCutcheon in 1956, is an architectural landmark that put Melbourne at the forefront of tall building design in the late 1950’s. It was Melbourne’s first fully glazed curtain wall skyscraper, more than doubling the city’s existing height limit, and the first building designed under new plot ratio determinations for city sites that now shapes Melbourne’s skyline. The building’s corporate-modernist style was representative of the new technologies of glass and steel and the new role of industry in construction, delivery, and repetitive efficiency. Entering the building, one passes a courtyard fountain by Gerald Lewers and, in the lobby, a large Inge King sculpture.

The Arup Melbourne office employs approximately 200 people. It is multidisciplinary, and includes Acoustic, Traffic, Mechanical, Electrical and Plant (MEP), Structural, Façade and other groups. My research has primarily been with the Structures group, the core activity of which is structural engineering. The Structures group comprises of 16 engineers and 4 CAD technicians.

Projects enter the Structures group via different means, and at different stages of development. They might begin as an Expression of Interest (EOI) process, through an invitation to enter a design competition with an architectural firm, as overflow from other Arup offices or as architectural concepts that are more or less resolved. Occasionally, for projects such as stadia or sports halls, the group is asked to design the building from scratch, with the architects choosing from a series of options. Arup Melbourne does not generate its own projects, but always works with an architectural firm. Arup works with both local and international architectural firms, typical projects being stadia, high rise buildings, educational facilities, bridges and other larger scale projects. The initial documents that Arup receives from these practices are generally 2D drawings and renders, and occasionally 3D models. The architectural firms that Arup Melbourne works with have varying levels of skill in 3D.

2 The Venice Bridge project, described in Chapter Six, represented the first architectural competition that Arup Melbourne has entered by itself. The office, similarly to all others within Arup, does not enter competitions as that would involve the practice in direct competition with its client base. For this reason, the Venice Bridge project was submitted using a nom de plume.
2.2.2 DIGITAL TOOLS WITHIN THE STRUCTURES GROUP

Unlike the typical architectural practice, where the roles of CAD operator and designer have converged, within the Arup Melbourne office there is a clear distinction between CAD technicians and engineers. Walking through the office, it is possible to pick one from the other by the computer equipment that is on, and not on, the desks. CAD technicians have two screens on their desks and a large size box under the desk. Engineers have a single screen and small size box, both on top of their desks.

The software used by CAD technicians/modellers includes Microstation™, Microstation Triforma™, Bentley Structural™ and Generative Components™, a parametric plugin for Microstation™. CAD technicians use this software to produce 2D digital files, 3D digital models, 3D PDFs, renders and 2D drawings. On average, one third of each project is done in 3D. The software used by structural engineers includes lightweight CAD viewing programs but is otherwise very different to that used by the CAD technicians. GSA™ (General Structural Analysis, Arup’s in-house structural analysis software) and Strand7™ are the primary structural analysis tools. Support for CAD and analytical software is available via an intranet and various skills networks, though which problems and knowledge are shared globally around the practice.

2.2.3 TECHNICIAN/ENGINEER INTERACTION

The primary role of the CAD technician within the office is to combine information that the architect and the engineer provides and produce an accurate, coordinated set of drawings. With the relatively recent ability to structurally analyse 3D structures another role has emerged. This is to assist the engineer in the development of the structural framing model.

In a typical project the engineer, while engaged in design exploration with the architect, designs a structural frame using butterpaper. This loose-fit solution will reach a point at which it is approximately 75% locked in, at which stage a CAD model will be made. The CAD technician introduces accuracy to the design, producing a structural framing model. 3D CAD tools allow this model to be constructed much more easily than they can within analysis programs, where you need to know what the start and end point is before you draw a line. This model is passed back to the engineer for analysis and design iteration. Over the last two years, Generative Components™ has been used by the CAD technicians on several projects to facilitate this iterative stage. This plugin to Microstation allows for the easy construction of the nodal, wireframe models needed for structural analysis and, as a parametric modeller, provides a variable-based method of controlling, for example, beam spacings that might change numerous times. When the structural design is finalised, the CAD technician will use the wireframe model as the basis for a building model, from which coordination with the architectural information is checked and drawings developed.
2.2.4 3D TOOLS
As Kvan has noted, the successful use of digital technology in professional practice is not so dependant upon the software and hardware provided as on “the compatibility of the software to the strategies of practice pursued by professionals applying the software” (Kvan 1995: 771). The strategies of practice employed by Arup Melbourne have been changing, and this has led to changes in the software and in who is using it.

Since at least 2005, Arup globally has been in the process of transitioning to 3D documentation. Simondetti & Brodkin’s internal review of the use of 3D digital tools within Arup globally, titled 3D Documentation Transition, identified that there was a direct competitive advantage observable in projects undertaken by Arup in which 3D documentation had been used, including lowered costs, more information, better coordination and marketing benefits (Simondetti & Brodkin 2005: 2). They stated that engaging 3D techniques promises to lead to the following future benefits for Arup:

- To extend our 3D skills to offer virtual construction services.
- To incorporate attributes in support of Facilities Management.
- To provide visualization images throughout the design process.
- To generate direct visual construction in areas such as structural steel.
- To gain a better command of complex geometries using parametric methods.
- To facilitate links between 3D analytical models and documents.
- To offer full coordination between trades
- To be the consultant of choice for clients and architects wishing to explore complex geometry as a design philosophy.

(Simondetti & Brodkin 2005: 4)

There are two primary drivers for the expanding use of 3D digital tools within Arup Melbourne. These are either to improve the process or to improve the end result – to do it in less time or to deliver a better product. Both within this report and throughout Arup offices generally, it is the first reason, being the economic benefits of using 3D digital tools to lower the costs of documentation and coordination and help avoid double handling and the loss of data that has received most attention. In some respects this may indicate the complexity of business overriding aspiration, but while important it does not exclude the recognition of significant alternative uses of 3D modeling tools. Simondetti & Brodkin, for example, include in their report instances in which 3D digital models were used to support applications other than drawing coordination and extraction. These include the use of Building Information Modeling (BIM), 4D construction planning, Regenerative (Automated) Modelling, Parametric Relational Modeling and Immersive Environments. Additionally, project examples demonstrating the use of 3D CAD and analytical

---

3 This is my observation, made having presented and discussed the use of 3D digital modeling tools at the Arup Berlin, Beijing, Dusseldorf, Hong Kong, London, Los Angeles, New York and Sydney offices.
models within design generation have been published in the Arup Journal, and the work of Cecil Balmond and the Advanced Geometry Unit is widely known. However, at the current time these investigations are regarded as secondary to finding efficiencies in the documentation process, and while the Arup Melbourne office has experience with using 3D models for documentation and coordination they do not have experience with these other applications.

2.2.5 CHANGING PRACTICES

During the three years in which I have been embedded within Arup there has been an increased use of 3D digital tools for documentation, analysis and geometrical manipulation which has led to several changes within the office. The first of these is that structural engineers are beginning to use 3D CAD tools themselves. Increasingly, the younger engineers are learning and using 3D CAD tools to complete simpler geometrical manipulation tasks that previously would have been performed within the analytical software General Structural Analysis™ (GSA) or by the CAD technicians. One reason for this is efficiency - drawing in GSA™ and other analytical software is a clumsy and time consuming process, while waiting for a CAD technician to make simple changes in a 3D model can be just as slow. In-house training in the modeling software Rhino™, a relatively quick and simple 3D tool, has allowed structural engineers to increasingly make these changes themselves. A second reason is that many architectural firms use Rhino™ to develop their 3D models, making it easy for the structural engineer to interrogate these models themselves. A third reason is that the acting group leader actively uses Rhino™ within his own structural design work. As the lead design engineer within the office he arguably occupies a position of role model for many of the younger engineers.

A second issue to emerge is that the push to 3D involves CAD technicians having to learn new high level skills. Finding or training people has been a difficult process, and skill levels are therefore varied within the office, ranging from technicians who have mastered 3D tools to those that only draft in 2D. Simply because of these resourcing factors, some projects are done in 3D and others are not. Complicating the situation, the break between 2D and 3D is by no means a clean one - 2D drawings remain a basic contractual requirement for building documentation. While many 3D CAD programs automate the production of these drawings reasonably well, they are not yet at the stage where they offer best practice in 2D drawing output - a complaint made to me by one CAD technician was that too much time needs to be spent making the output of a 3D process look like a traditional 2D, paper-based drawing. This changing state of affairs is crystallized in the terms ‘tracer’ and ‘modeler’, used throughout Arup. Tracer refers to technicians who have not learnt 3D tools, and are seen as ‘less able’. Modelers are those that have put a foot on the 3D ladder.

In summary, 3D digital tools are present and used at a relatively sophisticated level within Arup Melbourne, where they are leading to significant changes in the way the office practices. The current use of 3D modeling tools is primarily, but not exclusively, geared towards increasing the efficiency and coordination of documentation. The practice is aware of the potential for the use
of 3D modeling tools in other aspects of design but, at the time this research commenced, had not yet explored these avenues itself.

2.3 THE PARTICIPANT-OBSERVER ROLE
In conducting this research I have employed an approach termed participant-observation (Fig.7). This approach, which seeks to produce knowledge within a situated context of application (Lee et al 2000: 117), “recognises that theories are generated in context, influencing, and being influenced by a context of interactions as they are in the process of being developed” (Fook 2002: 81). In this section I outline the nature of this approach, the research instruments it is associated with, potential problems that have been identified and finally several different types of knowledge that can be accessed via the participant-observer approach.

2.3.1 PARTICIPANT-OBSERVATION
Participant-observation “requires the researcher to be in the field or present in the natural settings where the phenomenon under study takes place” (Maykut & Morehouse 1994: 72), and to engage in “repeated, genuinely social interaction with the members of the organisation under study” (McCall & Simmons 1969: 27). Via participant-observation, “the researcher becomes a practitioner in order to understand the practice situation” (Jarvis 1999: 100). There are several recognised research instruments available - these include direct observation, informant interviewing, document analysis, respondent interviewing, and direct participation (McCall & Simmons 1969: 27). Within my research, I have used three of these: direct observation, informant interviewing and direct participation. In parallel with the review of literature these instruments can support a process whereby “hypothesis generation, data gathering, and hypothesis testing are carried out simultaneously at every step of the research process” (McCall & Simmons 1969: 27). For the participant-observer, this process begins “with a broad focus of enquiry and through the ongoing process of observing and participating in the setting, recording what she sees and hears, and analysing the data, salient aspects of the setting emerge. Subsequent observations are guided by initial discoveries” (Maykut & Morehouse 1994: 69).

The participant-observer approach to research responds to the idea that “the disparities between knowledge and theory generated by professional researchers, and the ‘on-the-ground’ knowledge embodied in the daily experience of both practitioners and service users are widening” (Fook 2002: 81). Bridging this gap can provide a depth of understanding unattainable through other approaches (LeCompte & Goetz 1982: 32) but the researcher must be able to successfully straddle and reconcile academic and practice domains. A recipe of sorts is provided by Fook, who describes seeking “to minimize the

4 This mode of research has benefits and drawbacks. One significant but unavoidable drawback is that, in comparison to a more traditional approach to academic research, less time can be spent reviewing ’state of the art’ approaches before being immersed in the particularities of problem resolution. Within practice, the constraints of time, the uniqueness of projects and the requirement to begin investigatory work promptly may be viewed as typical conditions. For these reasons, leading and informative research from practice and academia may be left un-reviewed or unappreciated, and ‘from scratch’ approaches developed that suffer or replicate as a result. In contrast, within academic research one begins with the current state of the art as a basis, and aims to advance that method. With the aim of providing an account that is scholarly on the one hand, and evidential on the other, I have had to take a position with regard to this methodological dilemma.
influence of pre-existing formal theory; to maximise the number of perspectives available; and to maximise the fit between the method for accessing the experience, with the practice experience itself; and where appropriate, to include the perspectives of the practitioners/researchers” (Fook 2002: 85).

2.3.2 ISSUES IN PARTICIPANT-OBSERVATION

As Jarvis recognises, “straddling two occupations is a rather common occurrence – except, it appears on the surface, in research” (Jarvis 1999: 8). This can pose problems. Fook states that “practice does not lend itself easily to the requirements of traditional research as we see them – the need to measure and control variables, to make predictions, to be able to generalize our findings” (Fook 2002: 81). The very nature of practice “dictates that we are concerned with the specific” (Jarvis 1999: 84), which has the potential to make generalisation and replication problematic. One criticism of qualitative approaches such as participant-observation has been that they sometimes fail to adhere to the tenets of external and internal reliability and validity, and that their results can be consequently regarded as unreliable and lacking in generalisation (LeCompte & Goetz 1982: 31). While the criticism can be equally true in reverse – Bronfenbrenner argued that the overuse of laboratory studies in developmental psychology led to “the science of strange behaviour of children in strange situations with strange adults for the briefest possible periods of time” (Bronfenbrenner 1979: 19) - this problem of generalisation and replication should be briefly examined so as to guard against it.

In analysing the differences between qualitative research and its quantitative counterpart, LeCompte and Goetz point out that qualitative, ethnographic research often draws on the subjective experiences of the researcher and other participants (LeCompte & Goetz 1982: 32), and does so from within often unique contexts. Jarvis, for example, argues that practice is not an empirical phenomenon and should not be treated as though it were (Jarvis 1999: 83), while Fook privileges ‘accessing experiences’ over ‘obtaining data’ (Fook 2002: 83). In the case of this research, it has taken place within the Melbourne office of Arup, and has been influenced by and actively draws upon my presence and that of others who work within the office and for other architectural firms. It should therefore not be completely unexpected if it is not exactly replicatable in the offices of other engineering firms. Even within a practice, “every practice event is unique and ephemeral, and there is no empirical reality that can be carefully measured, checked and rechecked” (Jarvis 1999: 83).

To address these concerns, it is recognised that issues that impact upon validity, or the accuracy of findings, and upon reliability, or the replication of findings, need to be clearly identified and described (e.g. LeCompte & Goetz 1982: 32, Jarvis 1999: 83). This involves defining the context, the role of the researcher and others, and stating the processes of data collection. Provided that “they are sufficiently rigorous, well planned, and undertaken in the most professional manner, something of their reliability has to be admitted” (Jarvis 1999: 83). I provide such information within this Chapter.

To maximise the research input made available via my participation in live practice-based projects, I have prioritised input from the field over that generated via extensive literature review into all aspects of this thesis. Following the traditional academic model demands that a substantial proportion of the three year PhD period be devoted to such a review, leaving relatively little time to pursue the ethnographic and participatory investigations of practice that represent one key contribution of this thesis. The impact of this position upon my thesis is that a limited review of research is presented relevant to the specific digital techniques used in the projects reported in Chapter Six. These techniques cover parametric design, optimisation methods and approaches to analysis and information transfer, which are reviewed in a limited manner in Chapter Four. I recognise that in this area the thesis might be lacking, however assert that such an approach has enabled me to maximise the research input available through my role as participant observer within the available period.
2.3.3 ACCESSING PRACTICE KNOWLEDGE THROUGH PARTICIPANT-OBSERVATION

As a project participant, I have actively worked on ‘live’ projects that pass through the Arup Melbourne office and research projects within it. This participatory role has enabled project-based testing, which is one form of knowledge generation, but importantly has also opened up conversations with Arup employees that would not have occurred without this work at the coalface. As noted by McCall & Simmons, “the role assumed by the observer largely determines where he can go, what he can do, whom he can interact with, what he can inquire about, what he can see, and what he can be told” (McCall & Simmons 1969: 29). As a participant, I have had access to at least three kinds of knowledge:

**Tacit knowledge:** tacit knowledge is knowledge that is learnt through experience and embedded in a culture and difficult to articulate linguistically (Jarvis 1999: 48). Throughout the life cycle of a design project, architects rely heavily on their tacit design knowledge to support design decisions (Schon 1983). Nyiri (1988) in Jarvis (1999: 47) and quoting Feigenbaum and McCorduck (1984), writes that:

> One becomes an expert not simply by absorbing explicit knowledge of the type found in textbooks, but through experience, that is, through repeated trials, “failing, succeeding, wasting time and effort, getting a feel for the problem, learning when to go by the book and when to break the rules”. Human experts thereby gradually absorb “a repertoire of working rules of thumb, or ‘heuristics’ that, combined with book knowledge, make them expert practitioners.” This practical, heuristic knowledge, as attempts to simulate it on the machine have shown, is “hardest to get at because experts-or anyone else – rarely have the self-awareness to recognise what it is”. So it must be mined out of their heads painstakingly, one jewel at a time.

**Knowledge about experiences:** Fook identifies experiences as one of the key types of information sought by the researcher: “I find it more useful to talk about ‘accessing experiences’ rather than ‘obtaining data’, since the information we seek is the experiences themselves. Since these experiences already occur, it is more accurate to speak of accessing them in the most appropriate ways, rather than trying to collect something (‘data’) which does not already exist in the format we want it” (Fook 2002: 83).

**Working knowledge/Practical knowledge:** As defined by Jarvis, practical knowledge combines process knowledge, content knowledge and tacit knowledge and is legitimated in practice (Jarvis 1999: 47). Barnett, who refers to practical knowledge as working knowledge, refers to this convergence in the statement that “working knowledge is not only in work; it is what works” (Barnett 2000: 25). He claims that the working knowledge “most highly prized in the modern world is that which is produced in situ in the domain of work; that is, in settings that are systematic, collective, often large-scale and oriented towards production, profit and growth” (Barnett 2000: 16).
2.4 RESEARCH INSTRUMENT ONE: INTERVIEWS

As noted in section 2.1, much of the knowledge that is generated within practice is often not captured by individual firms or formalised externally as resources accessible to researchers. To overcome this obstacle, I have used a process of interview to gather information from experts in the field. The information generated via this process has been used within Chapters Three, Four and Five, and full transcripts of the interviews can be found in Appendix A1.

2.4.1 PURPOSE

The interview process was conducted with the purpose of gaining insight into how Practice experiences and understands problems connected to architect-engineer interaction during the early design phase. Taking a cue from Fook, who comments that “if I am aiming to theorize from practice as it is experienced by practitioners, I find it best to elicit their own descriptions of their practice, rather than study accounts constructed for other purposes” (Fook 2002: 84, her italics), the interview questions aimed to elicit concrete descriptions, based on experiences. This information has been used to understand the respondent’s point of view rather than make generalisations about their behaviour, and to verify and connect published theory to my immediate research question and Arup context.

2.4.2 PARTICIPANTS

Interviews were conducted with nine senior Arup employees, located in the Melbourne office and internationally (in the Arup London and Hong Kong offices). These employees included both those who led groups and those who worked within them. The Arup Melbourne participants included a Computational Fluid Dynamics (CFD) analyst, a lighting designer and a structural engineer who had worked on large collaborative projects for British airports and stadia. Interviewees from international Arup offices included two structural engineers from Hong Kong who had worked on large and well publicised projects in China and Hong Kong, an architect and an engineer from Arup’s Advanced Geometry Unit in London, an architect and group leader with an interest in stadiums, and a researcher with an oversight and dissemination role within Arup globally. Further details can be found in the List of Interviewees.

2.4.3 INTERVIEW TECHNIQUE

I have employed a semi-structured (Barriball & While 1994: 328) approach to the interview process. Semi-structured interviews can be understood as guided conversations in which broad questions are asked and new questions are allowed to arise as a result of the discussion. This approach differs from structured interviews, where a constrained list of questions is determined beforehand and the interaction between interviewer and interviewee is strictly controlled, and from unstructured interviews, where the interviewer listens but does not prompt. Semi structured interviews provide greater flexibility than is possible with surveys or structured interviews, and through the use of open questions allow a scope for themes and ideas to emerge during the
interview rather than being defined in advance. The central aim for each interview was to access the practice experiences and understandings of the Arup practitioners.

Prior to the interview, I had presented the context of my research and provided the participants with a list of questions. Additionally, the interviews usually took place after I had made a presentation of my work to the person in question, their group or their office. Where possible, the interviews were conducted within informal settings, preferably at the desks of the interviewees, and lasted approximately 30 - 45 minutes. The objective was to allow interviewees the opportunity to refer to their work, which some did but others did not. All interviews carried out, recorded and later transcribed by the author.

2.4.4 QUESTION LIST

Interviewees were asked four open questions. These were:

1. How and when do you communicate your ideas to architects?
2. What tools do you use in the conceptualisation and development of a design idea?
3. Where are problems of translation located within your digital workflow?
4. Where do you see integration occurring or failing to occur?

On many occasions additional questions arose during the interview, these took the form of ‘You said X a moment ago…can you tell me more?’

2.4.5 ANALYSIS

Interview transcripts were coded according to four themes which were emerging in the simultaneous undertaking of the projects and the review of literature. These were differing perceptions, joint and creative problem solving, communication and trust. The interviews were conducted, transcribed and analysed so as to extract ‘on the ground’ experiences, captured in interview segments, which related what I was reading about to what I was experiencing in the practice-based projects. Transcripts of the interviews can be found in Appendix A1.

Given the differing experiences, interests and levels of seniority of the engineers interviewed, it was not surprising that differing responses to each of the four questions emerged. On some points, most interviewees agreed. An example of this is that most engineers did not use 3D digital tools in the conceptualisation and development of a design idea, but rather methods like sketching. 3D models were likewise recognised as being highly effective in communicating design, however a common theme to emerge was that rather than 3D information, PowerPoint™ slides with graphs or 2D drawings were often used. An interesting note was that one of the engineers, an expert in CFD analysis, did not think he ‘designed’, leading to a discussion about what design was in his particular context.
2.5 RESEARCH INSTRUMENT TWO: PRACTICE-BASED PROJECTS

Being ‘embedded’ within a practice is to be confronted by the turbulence of that particular practice. This is particularly true of an engineering consultancy, where work is necessarily subject to numerous internal, external, social, economic and technological factors over which one person, let alone the practice, may have little control. Coming from architectural study and practice, in which either myself or the partners were the main drivers of design direction and decision making, I initially found this lack of control over the direction of a project very frustrating. In addition to this condition, the very structure of the design process, which is iterative and explorative, subject to personal judgements, often non-linear and given to taking the back roads rather than the highways, makes the careful, pre-emptive design of experiments practically impossible.

2.5.1 THE ‘COUNTLESS PROJECTS’ METHOD

The experimentation, then, has proceeded by surprise, with projects being taken up without knowing which would become substantial and which would be cut short - I view this as the only way that research in practice, carried out through ‘live’ projects, could have occurred. This being the case, it is no surprise that the two year long experimentation phase has been littered with false starts, aborted investigation and work that, while full of potential, never developed beyond the competition stage or was made mundane through factors outside my control.

Mixed into this mess is work that served no immediate research purpose, but was rather about getting the job done. None of this work is insignificant, nor completely out of the research ‘box’ – within such work problems are proven to be general in more or less extreme forms, and knowledge is developed about techniques and requirements that inform the more significant projects that are reported in Chapter Six – but of and by itself this section of work is made peripheral by its specificity and the shortness of its lifetime. Illustrated here (Figs.8–24) are some of the projects that did not precede, but which were by-products of the ‘countless projects’ method. Many are the result of the Melbourne office being asked to design structures, often for stadia or long span spaces, from scratch. The architectural firms involved in these projects have subsequently developed or disregarded the different design
options. All the projects demonstrate the basic advantage of visualisation that 3D CAD brings to the engineering design process, particularly in the design of geometrically challenging folded plate structures, as well as to the communication process. In reference to the research question, these projects show that performance criteria, whether addressing structural, spatial, environmental or movement aspects of design, are a significant driver of engineering design.

2.5.2 THREE APPROACHES TO PROJECTS

Given this condition it was necessary to develop three types of project based work so as to rigorously test the research question. The first of these types includes projects that were undertaken completely ‘live’, in which Arup Melbourne played the role of consulting engineer to an external architectural firm. These projects were undertaken with real world deadlines, cost constraints and the involvement of other industry participants. The second type consists of projects that began this way, but for whatever reason ceased as live projects. These projects were sufficiently interesting to be continued as studies that explored the implications of the ‘live’ work and continued to follow the logical progression of the project. The third type of projects was self generated and took advantage of the multidisciplinary nature of the Arup Melbourne practice. These projects sought to bridge the gap between different disciplines within the office, treating this situation as a microcosm of wider practice. By pursuing a mix of these different project approaches, the results of which are described in the experimentation chapter, sufficient ‘mass’ was gained to allow testing through projects to
Figure 14. Proposed canopy for Law Courts in Melbourne - option 1

Figure 15. Proposed canopy for Law Courts in Melbourne - option 2

Figure 16. Proposed canopy for Sports Hub in Singapore

Figure 17. Proposed canopy for Sports Hub in Singapore

Figure 18. Proposed canopy in Melbourne - option 1

Figure 19. Proposed swimming centre in China

Figure 20. Proposed canopy in Melbourne - option 2

Figure 21. Proposed swimming centre in China
become the significant focus of the research.

2.5.3 THREE PROJECT TYPES

In organising the projects, I have grouped them under three labels. These are 1. ‘Design\textsubscript{(Arch)} | Design\textsubscript{(Eng)}’, 2. ‘Design | Analysis’ and 3. ‘Design | Making’. Within each group one major project is reported, followed by several minor projects in support. The bar character that separates the two terms in each relationship denotes a concurrency operator, and indicates that the two processes execute in parallel. The section Design\textsubscript{(Arch)} | Design\textsubscript{(Eng)} reports projects that use a common language of geometry to link architectural and engineering design ideas through geometrical interpretation. The section Design | Analysis reports projects in which analytical tools have been used generatively to actively guide and synthesise design exploration. These projects include both optimisation and form finding processes. The final section, Design | Making, reports a project in which streamlining the design process around fabrication constraints led to a changed relationship between engineer, architect and steel fabricator.
SUMMARY
This Chapter has detailed the design of my research. It has briefly described the two research contexts, the first of these being the academic context of the Embedded Research within Architectural Practice program. The second context is that of the research setting, the engineering practice Arup Melbourne, with whom I have been ‘embedded’ for the previous three years.

I have explained my role of participant-observer within Arup Melbourne as one that requires the researcher to become a practitioner to understand the practice situation, and listed the kinds of knowledge that might be accessible using such an approach. Finally, I have detailed the two research instruments that I will be using in addition to the review of literature to further investigate architect-engineer interaction during the early design phase. These are the semi-structured interview and practice-based projects.
Chapter Three: Factors for Separation and Integration in Architectural and Engineering Design

3.1 COMMON ORIGINS: THE MASTER BUILDER
3.2 FACTORS FOR ARCHITECT-ENGINEER SEPARATION
3.3 ARCHITECTS VERSUS ENGINEERS
3.4 THE CURRENT CONTEXT FOR INTEGRATION
INTRODUCTION

Previously, men could be divided simply into the learned and the ignorant, those more or less the one, and those more or less the other. But your specialist cannot be brought in under either of these two categories. He is not learned, for he is formally ignorant of all that does not enter into his specialty; but neither is he ignorant, because he is “a scientist,” and “knows” very well his own tiny portion of the universe. We shall have to say that he is a learned ignoramus, which is a very serious matter, as it implies that he is a person who is ignorant, not in the fashion of the ignorant man, but with all the petulance of one who is learned in his own special line.

Ortega y Gasset 1932: 112

The current process of designing a building strings together a series of relatively discrete, sequential operations. Having been generated by an architect, a design concept is passed between aligned specialists who evaluate, refine, cost and implement. Within this sequential process, ‘tiny portions of the universe’, each speaking its own language, often collide yet integration has remained difficult to achieve within the current divisions of labour and though the available set of design tools and processes.

This chapter examines two such specialists, architects and engineers, introducing their characters and establishing the social nature of their relationship. Within it, I describe some of social and historical factors that have led to a ‘gap’ between architectural and engineering design practice, examine their current relationship and introduce the idea that interaction between architects and engineers might beneficially occur during the early design phase, at a time earlier than it currently does.

To establish the underlying differences between the disciplines, I firstly describe their common origin and divergence into distinct specialisations. Four significant areas of difference are examined: education, materials, representation and design process. Secondly, I examine the roles and attitudes that each discipline can assume when working with the other.

Then, to examine the contemporary divisional nature of architect-engineer interaction, I briefly describe problems of coordination, efficiency and responsibility that affect their relationship, and introduce the idea that architects and engineers might interact earlier and in a more collaborative manner. Lastly, I examine the nature of the early design phase, the suggested period at which this collaborative interaction should occur.
3.1 COMMON ORIGINS: THE MASTER BUILDER

The separation of building tasks into specialist skills for architects and engineers is a relatively new phenomenon. Prior to the Renaissance, the skills of architecture and engineering were invested in one person, the master builder, who possessed the integrative knowledge required for design and production (Hill 2005: 14) (Fig.25). This knowledge was precedent-based: during a long apprenticeship, generally as a stonemason or carpenter, the future master builder learnt the potentials and limits of materials, how they could be shaped and cut to enhance rather than compromise their natural strength, and which structures did and did not work. The master builder was “comprehensively and intimately familiar, at the same time, with the means by which his design could be brought to realization in actual stone and mortar” (Fitchen 1961: 10), the result being that “structural form, strength and stability, and architectural expression were inseparable and complemented each other” (Larsen & Tyas 2003: 30). This complex mix of intuition and experience was formalized in a set of concepts of geometric principles and proportion, and rules of thumb that were usually jealously guarded leading to the formalization of specialists as ‘guilds’ This period can be identified with the formation of the disciplines that we are familiar with today.

On site, local materials were used, construction was labour intensive and there were ongoing collaborative relationships between designers and craftsmen. There were generally few drawings or models, instead verbal communication and full scale site layouts were employed. This way of working, and particularly the dependence on verbal communication,

---

1 The Greek root of architect is arkitekton, meaning master builder.
required the master builder to be continuously present on site and limited the number of large scale projects they could work on to, typically, one project at a time.

In the 15th century, design and construction skills began to separate into distinct disciplines or guilds. Scholarly knowledge was placed on par with practical experience, and drawing was developed as an effective technique for the remote communication of design information. These drawings were very imprecise, and their deployment involved much negotiation between site and design intent (Robbins 1997: 16); the effect, however, was the withdrawal of the architect from a hands-on contribution on the building site to something akin to a professional office and a widening gap between the designer and builder (Fig.26).

Projects during this period were generally coordinated by the client, who supervised a team consisting of a creative artist (a goldsmith, painter or sculptor), an architect for technical knowledge and site supervision, and a master builder for construction (Barrow 2001: n.p.). Up until the industrial revolution, the design side of this equation was relatively stable - the term architect covered a wide range of activities, including that of structural engineer² (Larsen & Tyas 2003: 29) – mainly, it has been argued, because of the limited variety of construction materials and techniques (Barrow 2001: n.p.).

3.2 FORCES FOR ARCHITECT-ENGINEER SEPARATION

During the 19th century, the idea of unity of knowledge gave way to that of specialisation (Fig.27), and the generation of knowledge through a divide-and-conquer strategy has subsequently underpinned much of the development of the growth of the disciplines and modern science. Specialisation brings the benefit of “allowing specialists within a discipline to refine theories, methods, and technologies and push outward the bounds of knowledge within that field” (Seipel 2005: n.p.). In practice, it allows “average professionals to collaborate and often achieve results almost as great as those of the geniuses of the past” (Salvadori 1991: XV). By the year 1987 there were 8,530 definable knowledge fields (Crane & Small 1992: 197) of which approximately 50 occur within the construction industry (Tombesi 1997: 19). The number of specialist disciplines tends to support the idea that we live in a society that demands specialisation as a prerequisite for competence.

Within a specialisation, practitioners “share basic assumptions about the nature of the world, beliefs about what constitutes an interesting question for study, methods for generating and analysing information, and rules about what constitutes evidence or proof” (Seipel 2005: n.p.). As specialists, the domains of architect and engineer have become distinct² and, while architecture has until recently maintained a relatively stable state, the number of specialists within engineering has dramatically increased: structural, mechanical, façade, lighting, acoustic, fire and pedestrian engineers are today required to contribute their particular expertise on the typical large scale building.

² Disagreeing with the concept of the master builder and disciplinary evolution, Billington argues that there is no historical relationship between modern architects and structural engineers, but that modern engineering developed “parallel but independent to architecture”, and that the two disciplines have never been under one roof.
Echoing the idea that the number of stars in the sky is dependent on the strength of the telescope, increasing refinement in instrumentation, in particular computer usage, has rendered new things measurable, and new specializations have emerged to deal with them (Lubkeman 1992 n.p.). The rapid evolution of technology is further driving increased specialization in all parts of the building and construction industry and engineers have “become so highly specialized that they are seldom able to understand the work of colleagues in a different field of engineering” (Salvadori 1991: XIV). Similarly, construction and procurement have become increasingly specialized; currently “steel and concrete are not only manufactured by different industries but also fabricated by different interest groups” (McCleary 1991: 53). Continuing a trend noted by Ruskin, that “we are always in these days endeavoring to separate the two; we want one man to be always thinking, and another to be always working, and we call one a gentleman, and the other an operative, whereas the workman ought to be often thinking, and the thinker often to be working” (Ruskin 1853: 47), ‘working’ is increasingly separated from the world of ‘design’ thinking. Intermediary consultants, such as project managers and quantity surveyors have become commonplace, and consultant firms are emerging as ‘keepers’ of the digital models of buildings and managers of associated risk and responsibility.

There are many factors that have led the architectural and engineering professions to gain distinct disciplinary skills. While it is outside the scope of this chapter to examine them all in detail, four particular forces for separation that illustrate these differences are education, materiality, representation and design process.

3.2.1 FORCES FOR SEPARATION: EDUCATION

Responding to the separation of labour and therefore of educational systems, écoles and polytechnic schools emerged in late eighteenth-century France. The first engineering school was founded in Paris in 1720, and the first separate civil engineering school, the École Nationale des Ponts et Chaussées (the French School of Bridges and Roads) in 1747. An emphasis on the use of new materials (Billington 1991: 4) converged with the appreciation for an economy of means and methods, and at this point, “the struggles between builder and decorator, Ecole Polytechnique and Ecole des Beaux Arts, began” (Benjamin 1969: 165).

While engineering schools coexisted with architectural ones and academics crossed easily between them (Holgate 1986: 102), the consolidated separation of educational systems in modern times has meant that “the differing roles attract people with differing personalities and the educational training reinforces their disparities” (Holgate 1986: 93). Belcher notes that in their education, architects are “trained to think down, to synthesise a global solution [while] engineers are trained to analyse available data and to solve problems bottom up, following a more systematic process towards a single ‘best’ solution” (Belcher, in McLeod 2004: 27). A blunter description of the differences between architectural and engineering students is that “they belong to different breeds of the human species” (Salvadori 1991: XIII). One breed accepts the dictates of science, applies them as a means to address practical problems, and often “really believe that science explains physical reality” rather than describes it. The other breed is open-minded,
adventurous, curious and less interested in the practicalities of life - at least “until they become successful architects” (Salvadori 1991: XIV).

A contemporary example of the educational divide is provided by Maher and Burry (2006), who describe the current educational environment at the Royal Melbourne Institute of Technology. At this university, the faculties of engineering and architecture belong to different schools which are located in separate faculties. In describing their own attempts to address the widening gap between the disciplines they observe that “divisions are easily established and then maintained” (Maher & Burry 2006: 202). The educational divide is, however, not universal. Other programs, such as the Civil Engineering program at Bath University (UK), feature group project work between collaborating civil engineering and architecture students to introduce them to inter-disciplinarity, and in many European countries architecture is a specialty within the department of engineering and vice versa.

3.2.2 FORCES FOR SEPARATION: MATERIALS
Beginning in the 1800s, a series of material advances in iron, steel and later reinforced concrete allowed the construction of larger structures and longer spans (Luebkeman 1992B: n.p.). These new materials challenged existing practices in several ways. Culturally, they “broke so radically with conventional taste, they were rejected by the cultural [architectural] establishment” (Billington 1983: 15). Technically, they required the development of new techniques. While engineers explored these new materials and the larger structures in which they were used, architects tended to maintain the rule of thumb methods sufficient for small scale timber and masonry buildings (Holgate 1986: 102). McCleary notes that this difference continues, observing that architects “tend to favour those materials with which their profession has had a long experience”, whereas engineers “are more likely to use materials invented in the nineteenth and twentieth centuries” (McCleary 1991: 41).

The most significant impact of the new materials, however, was that they freed form from structure. McCleary notes that the introduction of steel in particular had a significant impact on the way that architects and engineers design, where increasingly the “collaboration and synthesis that originates in the confluence of the dimensionality and directionality of the architect’s space with the engineer’s structure” (McCleary 1991: 46) was no longer necessary. In a similar vein, Balmond has reflected that “there was a time when the assumption for architects was of a linear space, with an explicit assumption that structure was inherent in architecture. Too a large extent… architecture would go separately, and engineering could follow” (Balmond 2004: 143).

3.2.3 FORCES FOR SEPARATION: REPRESENTATION
As Peters states, “engineers think primarily in mathematics and architects in visual language” (Peters 1992: 1). The introduction of new materials to the architect’s and engineer’s repertoire required a

3 Examples include ETH Zurich, Milan Polytechnic and Barcelona UPC.
higher level of precision and calculation because their physical properties were not completely understood (Holgate 1986: 102). Initially alongside drawing, engineers developed algebraic methods for understanding the behaviour of these structures under an increasing number of different loads. Through mathematical representation engineers were able to engage with many concepts that could not be adequately addressed by drawing, and where attempts to understand them through drawing were likely to be fundamentally flawed (Robbins 1994: 288). As Rice observes when interviewed by Robbins, “as buildings get lighter . . . environmental loads like wind, snow, earthquakes, temperature effects, and such become increasingly important. These kinds of loads have little to do with the general shape of a building. Unlike a gravity load, which is visible, they are not” (Robbins 1994: 288). While mathematical representation is now central to engineering, Banham has stated that “being unable to think without drawing remains the one true mark of one fully socialized into the profession of architecture” (Banham 1990: 25), an observation that coincides with that of Lawson (Lawson 2004B: 53). 20 years later my observations from practice suggest that this statement still holds true.

3.2.4 FORCES FOR SEPARATION: DESIGN PROCESSES

Divergence in educational systems, familiarity with materials and modes of representation are matched by different approaches to design. Cross and Roozenburg (1992) have compared what they describe as the more linear ‘consensus model’ of engineering with the spiral structure that has emerged in architecture and industrial design. They note that the engineering model, influenced by electrical engineering, emphasizes the sequence of stages through which a project should progress (1992: 325); two feature of this model are a linear progression from the abstract to the concrete, and the splitting of complex problems into sub problems which are solved then re-synthesized (1992: 327). In contrast, architectural models of design emphasize the thought processes that the designer should perform and use a pre-structured ‘proto model’ as the primary generator which frames further exploration (1992: 330).

While many of the key figures within the design methods movement have since disassociated themselves from the field (e.g. Alexander 1971 & Jones 1977), differences in the design process remain. As Haber has noted, “conceptual design for an architect can be very abstract, with content that might be more poetic than geometric. Conceptual design for a structural engineer tends to be more concrete in nature - the choice between an arch and a suspension structure, between concrete or steel” (Haber 2000: n.p.). Identifying different attitudes to ambiguity as a key difference between architects and engineers, Haber continues to observe that “engineers, particularly those with a mathematical bent, expend a great deal of effort to eliminate ambiguity from their terminology and methodology” (Haber 2000: n.p.), and prefer problem statements with only one solution over those with multiple solutions: “reliable analyses and designs that can be executed in a predictable manner” (Haber 2000: n.p.). For architects, in contrast, “the tension and richness provided by ambiguity and multiple meanings is a hallmark of high-quality architectural design”

4 During the 1960s, the “design methods movement” aspired to ‘scientise’ design by basing the design process and by extension its outcomes, on objectivity and rationality.

5 While the arguments that I make within this thesis necessarily focus on the possibilities of the computer to support design exploration, it is important to note that the primary design task of the engineer is to develop an intellectual picture of how a structure or environmental strategy should work. Computational tools are one tool amongst many that can aid in this process.
(Haber 2000: n.p.). From my work at Arup, I have observed that generally, the further the architectural concept is progressed the more the engineer initially engages in a process of extraction – extracting possible structural diagrams (representing potential solutions) from the more complicated architectural information they have been given.

3.2.5 SIMILARITY

As Peters has noted, “it is curious that there should be two fields devoted to the same activity, namely, the erection of structures” (Peters 1987: 11), and it would be more curious if there were not some areas in which the disciplines were similar. The collaboration of engineer Fazlur Kahn with Skidmore Owings Merrill architects (Fig.28) (Billington 1983: 244), the exploration and synthesis of structural thinking of Antoni Gaudí (Fig.29) (Burry et al 2004), and the synthesis of sculptural thinking and structural optimisation of Sergio Musmeci (Fig.30) (Mostafavi 2002), suggests that for all the difference there are productive overlaps.

Two areas of overlap can be discerned in McCleary’s statement that “for the architect, design begins by considering the humanisation of space – the main concern is dwelling; for the engineer, design begins by answering to the properties of materials and the logic of structural mechanics – the focus is on structuring” (McCleary 1991: 45). Similarly, both architects and engineers are active designers, though the processes by which they design differ. As Billington has observed, different engineers “will find radically different solutions based on their own vision of what is appropriate” (Billington 1991: 14), as will different architects. Similarly, geometric form plays a significant role in both design processes, though structural designers...
“See forms as the means of controlling the forces of nature to be resisted; whereas architectural designers... see forms as the means of controlling the spaces to be used by people” (Billington 1991: 15).

Ankrah and Langford (2005) reveal further similarities in their statistically-based comparative study of the organisational cultures of architects and contractors (Fig. 31). They find that both architects and contractors “tend to adopt marginally decentralized approaches to management and decision-making with employees being allowed to participate in problem-solving. Employees are considered important to organizations and there is reasonable interest in their well-being. Both groups organize tasks so as to be performed by teams of employees, and administrative tasks are not seen as being as important as the other professional tasks” (Ankrah & Langford 2005: 604). In the uptake and use of new technology, “there is also a general readiness in both samples to adopt new and innovative technologies” (Ankrah & Langford 2005: 604).

Figure 31. A comparative study of architects and contractors (Image source: Ankrah & Langford 2005)
Ankrah and Langford’s study shows that in many respects, architects and contractors behave in a similar manner and have similar organisational cultures. This is observable in the categories technological readiness, appreciation of skill, calibre of employees. Areas of distinct difference occur in the need for recognition, departmentalisation, the nature of tasks and the tolerance for ambiguity. Interestingly, although beyond the scope of this research, the study finds that while the power of skills is comparable across the industry, the power of relationships is higher for contractors. In analysing their data, Ankrah and Langford (2005: 605) observe that:

“although there remains much in common, the various participants – and in this case architects and contractors – bring to this team different ways of thinking, different attitudes, practices and approaches to work...these differences imply a likelihood of conflict at the interface level where the human interaction elements come into play, and this has the potential to detract attention from either schedule or budget”

3.3 ARCHITECTS VERSUS ENGINEERS
In contemporary practice, the roles assumed by the architect and the engineer and the structure of the relationship that emerges can vary greatly, being dependant on the nature and working relationship of those involved and the constraints and opportunities of any particular project. While this is understandable, more confusing is the “tradition of antagonism [that] has arisen between the professions of structural engineering and architecture this century” (Peters 1991: 23). Understanding the relationship needs to take place at two levels – firstly, by entering the slanging match of disciplinary contrasts, and secondly by examining the different roles that one discipline can take with relation to the other.

When Billington states that “the prototypical engineering form – the public bridge – requires no architect. The prototypical architectural form – the private house – requires no engineer” (Billington 1983: 14), he raises the idea that each profession only achieves optimal results in the absence of the other. Is this actually the case, or is there more truth to the old joke that if an architect builds a building without an engineer, it falls down, but if an engineer builds a building without an architect, it will be demolished? One issue central to the contrast of disciplines is that that there is a mutual feeling of inferiority towards the expertise of the other (Peters 1991: 23). Candela (Candela in Faber 1963: 14) provides the architectural view:

“The second design phase consists of a tremendous struggle between the structural engineer and the architect...the architect wants to maintain his preconceived idea, but has no weapons to fight against the scientific arguments of the technician. The dialogue is impossible between two people who speak different languages. The result of the struggle is always the same: science prevails and the final design has generally lost the eventual charm and fitness of detail dreamed by the architect”
As Schwitter notes in reflecting on the practice of Buro Happold engineers, “quality demands that it remains a wholly engineering discipline practice. The fence between the two professions is important because it is absolutely critical that you know what is inside your fence” (Castle 2002: 68). The feeling that drawing closer to engineering will limit the architect’s artistic expression is linked to what Billington describes as the mis-held view that “engineering, being an applied science, merely puts into practice the ideas and discoveries of the scientist” (Billington 1983: 8). Billington has been instrumental in arguing for the art in engineering; however others have not been so keen to attribute engineers with a design sensibility. Le Corbusier, despite the Modern Movement’s embrace of engineering, characterised engineers as “healthy and virile, active and useful, balanced and happy in their work, but only the architect, by his arrangement of forms, realizes an order which is a pure creation of his spirit… it is then that we experience the sense of beauty” (Le Corbusier, in Mallgrave 2005: 256). According to this view, engineers who are mere servants of science do not necessarily think independently. Thus, as Collins observes:

“The architect’s constant plea for greater collaboration and co-operation between architects and engineers is thus primarily a slightly petulant demand that the engineers shall shake themselves free from yesterday’s outmoded structural systems, and calculate wholeheartedly the novel forms sketched out for them, even though these at first appear unstable (as indeed they are often, paradoxically enough, intended to appear)” (Collins 1960: 31)

The difference in professional expertise is also located in the sensibilities that each profession brings to the design of form. Rice distinguishes between the creative responses of architecture and the inventive responses of engineering (Rice 1998: 72). Creativity, in Rice’s view, stems from personal considerations - the architect is employed to give a personal solution, while in being inventive the engineer objectifies the problem. For the engineer, the design search and solution might be motivated by wanting to exploit the material properties of concrete, for example. This view does not seem to be shared by the broader public however - a Harris Poll sponsored by the American Association of Engineering Societies and the Institute of Electrical and Electronics Engineers -USA found that “only 2 percent of the public associate the word ‘invents’ with engineering; [and] only 3 percent associate the word ‘creative’ with engineering” (Bellinger 1998: n.p.).

### 3.3.1 ROLE CHARACTERISATION

While many architects tend to see the engineer as a willing servant (Holgate 1986: 91), there are several distinct roles that they can play within the design process. Guy Nordenson has labelled these roles as those of ‘technician’, ‘artist’ and ‘collaborator’ (Nordenson 2000: 37). The technician solves the problem given to them by the architect, the artist develops a distinct style out of the materials and methods of structure, and the collaborator’s work is “manifested in a sensibility rather than a style” (Nordenson 2000: 37), and may be unnoticeable. From an architectural viewpoint, MacDonald (1997: 25) has developed equivalent categories which cast the architect’s possible relationships to the structural engineer as ‘structure ignored’, ‘structure accepted’, ‘structure
symbolized’ and ‘structural hi-tech’.

What occurs when these roles collide? Salvadori has argued that problematic encounters occur not in the extreme cases, when the great meet the modest, but rather in the majority of cases when the modest meet the modest: “what transpires on these occasions is that the two professionals not only have different kinds of minds and, hence, different approaches to the same problem, but also have a real problem in communication: they talk different languages and do not understand each other” (1991: XIV).

To appreciate the outcomes of such a meeting, we can turn to Ove Arup’s taxonomy of potential design outcomes that occur in the wake of poor architect engineer interaction:

- **Starved Designs**: deprived of the benefit of the technical knowledge which could have improved it, had it been considered. As when engineers are called in too late to an already frozen design, or when the designers simply do not know their jobs or do not take the trouble to consult those who do

- **Forced or Lopsided Designs**: when put in a straight jacket of architectural formalism or structural acrobatics or client’s prejudice, disturbing the balance of priorities

- **Loose Designs**: when no proper synthesis is achieved for lack of effort or collaboration, hardening into:

- **Split Designs**: when the design is being handled by different authorities who barely communicate with each other

- **Pinched Designs**: due to economic stingency, when the ship is spoilt for a ha’p’orth of tar

- **Patched Up Designs**: when the brief is altered or added to by clients, or the architect has a better idea or additional information comes to hand which is somehow tacked on to the design without taking the only course which can assure a proper digestion of the new data: that of starting all over again

  (Arup 1985: 8)

### 3.3.2 A PERSONIFICATION

Ove Arup’s 1968 self reflective characterization of the structural engineer synthesizes much of that discussed above (Arup 1985: 2). Arup provides his characterization with a name, Ernest.

---

6 For example, when the engineer technician meets the architect who ignores structure, or the engineer artist meets the architect who celebrates structure.
A difficult, critical and outspoken and probably foreign individual, “he was no artist, and he decided to become an engineer… a good engineer; because he chose this way also in search of an opportunity for artistic fulfilment” (1985: 2). Arup aligns the engineer with an art or craft tradition, rather than that of science, specifically stating that engineering is not a science, but rather uses the general laws developed in scientific study to solve the practical problems of designing exciting structures. The art, or design, is in creating “a synthesis of means and ends” (1985: 2), a creative activity for which there are many solutions; good, bad or indifferent.

After an education that “didn’t get him very far when it came to actually designing” (1985: 3), Ernest developed a particular interest in a new material technology, reinforced concrete. He began to work with some of the pioneering architects of the Modern Movement, who enthused about engineering and the functional use of structure, but their aesthetic use of reinforced concrete ignored the fact that other materials could perform more economically. They knew “next to nothing” (1985: 5) about reinforced concrete and, no longer master-builders, had lost their connection with industry, practical building and building costs. Working with architects meant that Earnest had to accommodate modifications to the engineer’s aim of designing a structure – finding the most direct means of getting force to the ground – by either hiding the structure, or arranging it in response to needs other than those of structure and economy. Any move away from routine design and standardized techniques was limited by the fragmented nature of the building industry and the complete separation that had developed between design and construction – new methods needed the collaboration of others, particularly the contractor, who was “reluctant to plunge into the unknown” (1985: 6) and provide cost information about a design that was not yet finalized.

SYNOPSIS
As disciplines, architecture and engineering have developed different structures of value and distinct ‘brands’ of professionalism. Approaches can converge or conflict, as each discipline brings specialist knowledge not possessed by the other to the interaction. The need to work closely together yet simultaneously maintain a fence around their respective professional knowledge has resulted in friction. It has also generated stereotypic generalizations such the ‘daringness’ of the architect, who generates novel and perhaps unimagined responses to complex problems, and the ‘dependability’ of the engineer, who is contracted not to take risks but rather for the ability to calculate an outcome to a high degree of precision (Francisco 2004: 13). The gap that currently separates architects and engineers is located not in any single place but is between two equally complicated cultures; this will be further explored in Chapter Four.

3.4 THE CURRENT CONTEXT FOR DESIGN INTEGRATION
Despite some notable examples, architects and engineers remain distinct domains of practice only weakly linked through an overall common cause (Matthews et al 1998: 124). Interaction between them is generally conducted in a sequential manner, where design information is passed ‘over the
fence’ between the architect, who has designed a form, to the engineer, who will evaluate it, and then onwards to those who build and run the building. As one moves through this sequence, it is inevitable that miscommunications and conflicts will arise, and the further the design process progresses, the more difficult and costly change becomes.

3.4.1 THE CURRENT PROCESS

As Belcher notes, the “general perception is that the architect designs and the engineer solves – naturally in this order” (Belcher, in McLeod 2004: 27) (Fig.32). The interaction between the two disciplines typically takes the form, as Flager and Haymaker have identified, of three

![Diagram](image-source)

Figure 33. Three stages of design - generate, analyse & decide (Image source: Flager & Haymaker 2007)

iterative steps: generation, analysis and decision (2007: 626) (Fig.33). Within the generation phase, the architect creates a design that responds to the requirements of the client using 2D or 3D representations. The building design is then passed to the engineer for analysis, and the analytic results used to complete or alter the building design. The architect and engineer then meet to decide whether the design still correlates with the initial design concept.
This model is limited in several ways. Architectural and engineering design generation and refinement responds to a set of criteria that is limited to their own domains. The process presumes a limited interchange of information (Matthews et al 1998: 124), and there is little iteration. Typically “less than three such iterations are completed during the conceptual design phase” (Flager & Haymaker 2007: 626).

Affecting the process are issues of coordination and responsibility. Design information typically gets communicated between architects and engineers as paper-based drawings or as 2D digital files (Bernstein 2005: n.p.). As information flows back and forth the risk of that information, and the ideas that support it, becoming uncoordinated increases. This is because the 2D representations do not match those needed by many of the designers and makers involved in the process, who with difficulty extract the often ambiguous information they need from the drawings.

Even when a 3D model has been generated by a designer, its use to produce coordinated drawings is rare. Issues of risk make it unlikely for the model itself to be provided to other participants. Often, the 2D information taken from one 3D model forms the basis for another 3D model, implying significant reworking and double handling of information. As Bernstein has commented, “between each phase of development, those wonderful technology tools are used to reduce project information to its least useful form: paper. The result is a loss of quality and signal strength” (Bernstein 2005: n.p.). The back-step or loss of design information that occurs at each transition is crystallised in his sawtooth diagram (Fig.34), which provides evidence for Beck’s observation that “just because the pipeline gets bigger, permitting more uncoordinated information to flow through it faster, does not mean that the value gained within the process has increased significantly, nor are the individual participants necessarily any better off” (Beck 2005: n.p.).

As Laing and Kraria observe, “such routines arise in building design because designers find collaboration among themselves difficult to control”, with the difficult task of integrating the different aspects of design work “ultimately falling upon the construction manager or the contractor” (Laing & Kraria 1994: 235). Imprecision in communication, and the perceived need to transfer rather than mitigate risk, has encouraged project participants to work within clearly defined boundaries. “To protect the various partners from litigation, strict boundaries have been defined for the scope of responsibility each party takes on, and the flow of information between parties” (Shelden 2006: 37). While the “conventional ‘throw it over the fence’ organizational model does at least have the advantage of giving fairly hard and fast rules about responsibility and the compensation for accepting it” (Chaszar 2006: 216), it is being challenged by 3D digital information that is inherently precise (but which can still be plain wrong) but which is limited by existing contractual practices: “the seamless flow envisioned by digital working methods requires much greater flexibility (and perhaps agility) from the participants” (Chaszar 2006: 216).
Figure 34. The fall-off of design information between each phase of design development
(Image source: Bernstein 2005, adapted)

Figure 35. Existing Design Process (Image source: Pressman 2007, adapted)

Figure 36. Preferred Design Process (Image source: Pressman 2007, adapted)
3.4.2 COLLABORATION AND 3D TOOLS

With the increased use of digital tools and design techniques leading to increasingly complicated designs that need to be realized in the context of complicated constraints (including project timescales and budgets that have become more limited), the model described above is recognized as wanting.

Within the wider context of digital design within architecture and engineering there is a growing realisation that the interaction between architects and engineers should be pushed forward, into the early design stage (Figs.35 & 36). During this time the benefit of closer interaction, being more insight into the developing design, is greatest and the costs minimal. As Bernstein notes, “it’s crucial that this insight be available as early in the creation process as possible—that’s when there is the greatest potential to affect the building’s cost and function with the least cost in dollars and time” (Bernstein 2005).

Mr. G identifies these benefits when he states that:

The earlier in the design process the more effective it tends to be. There is a huge market for Arup, for instance, to develop Day 1 tools. You have to build up those interactions, to find a way in to that process, because it is not normally led by the engineers anyway. It’s got to be something that the architects can work with - as an architect I would expect you to be developing tools that you use from day 1, solving design using them, and that let you collaborate with others. For me, it is very much the earlier the better.

I am keen to say we start right here. The other thing about starting right at the beginning is that it must be the most beneficial point in the thought process. The majority of the key decisions are taken there – do we put our effort into creating something at the end here, which way do we put a screw in the hole, or do we do it here? I haven’t seen enough people coming back to the point of conception, they are generally at a point beyond that.

Mr. G

For architects and engineers, “this means a shift forward in the design process where engineers are asked earlier for their input in the design solution” (Achten 2002: 1). To work in such a way is seen to require a particular process, collaboration; Salvadori, for instance, states that “collaboration should be the basis of work between the architect and the engineer” (1959: 17) and Kolarevic has described the “collaborative quest” of architects and engineers (2005: 200). Achten observes that “in collaborative design processes, they [engineers] are expected to act earlier in the design process, where ideas are still in their formative phase” (2002: 9), and in reviewing the expanded role of the engineer in current practice, Rappaport states that “this paradigm has emerged through intense collaboration” (2007: 90).

---

7 ‘Insight’, of course, is found across the domains of architect, engineer and contractor, and more informed design exploration would ideally involve the early input of all these participants.
Matthew Fuller has described collaboration in practice as like playing the surrealist game Exquisite Corpse (Fuller 2005). In this game, each player draws one part of a body onto a piece of paper that is passed around the group. The paper is folded so that each body part is separate and the rest of the body remains unseen - the key to continuity between parts are the small registration marks that cross the folds. As Fuller argues that technologies can provide similar entry points, and that finding and working at these entry points offers the means to engage, so too have many observed that 3D digital tools can provide a means for enabling architects and engineers to collaborate (e.g. Achten 2002, Kolarevic 2005, Maher et al 2003, Shelden 2006).

In current practice, however, the use of 3D digital tools remains “mired in the past” (Kvan et al 2003: n.p.). As noted by Burry et al (2000: 135), the D in Computer Aided Design might stand more accurately for drafting, as the tools have been predominantly used for representative rather than synthetic purposes. A contributing factor must be that the developers of CAD systems have limited models of users and the problems of use (Henderson, 1998: 139); Achten et al observe that vocabulary of CAD systems (vertices, lines, planes, volumes and operations) is adequate to describe, but not support, design. (Achten et al 1996: 1). The focus on documentation in particular has made most CAD systems today productive only when the design is more or less completed. However, as Lawson correctly observes, “drawing is not an end it is a means to an end. Concentrating only on it will not enable us to make real progress with computer aided design” (Lawson 2005: 389).

The role of digital tools, and of architects and engineers themselves, in facilitating processes which engage both parties in earlier and interdependent design exploration are not as yet fully determined. As interview subject Mr. E describes, the points of intersection change from project to project:

The points of intersection are completely grey area, and it’s up for grabs. Both architects and engineers are interested in that area. [The architect is] interested in that from an architectural side, we’re interested in that from an engineering side. It’s also a commercial position - basically there is a market there for new ideas, new geometric concepts, new principles that go far from the traditional engineer architect relationship. It’s a much bigger area, increasingly bigger. The tools are more powerful and the architectural horizon has folded back.

There is no typical way of working with the architect; there are only ad hoc ways that suit specific projects. We work with different architects who have different agendas and different expectations. We do engineering, but we do engineering plus. We give very good ideas about how to construct the form, to model the form and analyse the form. In other cases we come up with entire geometric generators. If you work with an architect who is very formalistic then there’s almost no argument, they will start with an idea of what the building should be and then it’s a matter of making that building slightly smarter. Of more interest to us is when we can influence the concept, because then we go into something different that moves you away from that preconceived stylistic idea of what’s good or bad.
3.4.3 THE EARLY DESIGN PHASE

Interdependency, the concept which I will extract from that of collaboration, and the use of 3D digital tools will be more fully examined in the following two chapters. Before doing so, however, it is necessary to describe the early design process. Based upon my theoretical research and practical experience – within Arup Melbourne and in architectural practice – I view design as a process of working from abstractions to specifics with increasing commitment and precision; that is, a process of incremental formalization. This view coincides with that of Gross (1996: 54) and Aish (2005: 62), who describes this process as a movement from intuition to precision. Making the early design phase difficult is that part of the process involves finding/determining what is needed (Gero 1998: 165).

The early stages of the design process includes the conceptual and schematic design phases, and is characterized by competing requirements, ill-structured problems* and the ongoing formulation of geometry, materiality and other design information (Marsh 1997: A-15). From my experience within Arup Melbourne, I have observed that the early design stage is an iterative, exploratory phase consisting of ideation, articulation and evaluation, and traditionally has been supported by quick sketches, simple geometric analysis on a drawing board, simple hand calculations and some 2D CAD. The designer asks questions (Rowe, 1987: 41), and uses the new information provided by the answers to guide the design process.

A main criteria for these tests is speed (Marsh 1997: A-15), and their purpose is to “obtain better or more accurate information about the proposed building project without being obliged to undertake the work entailed in producing a detailed design and accompanying documentation at an very early stage” (Crawford 2004: 2) – to inform a design that is still in formation “within a context of incomplete and imprecise data” (Akin, 1998: n.p.). Using the techniques described in the preceding paragraph, the extent to which testing can occur is very restricted. The use of CAD within this phase is, as Fekete has observed, “judged simply not worthwhile” and “in the context of architectural design today, particularly during its early stages, the benefits of cad are far from being that significant” (Fekete 2003: 246). He states that “the main objections on the designer’s part towards such practice concerns the difficulties associated with modelling the loosely defined information so typical of the early design stage” (Fekete 2003: 246).

While the designer’s ability to evaluate proposals is limited by the incompleteness of data, many of the decisions made in the early design stage are critical to the ongoing definition of the design and the outcome. “Building projects generally follow the Pareto Principle or 80:20 rule, where 80% of the decisions

---

8 Simon’s definition of structured problems is as follows (all other problems can be defined as ill-structured):
1) All initial elements that enter into the solution of the problem are known and described for example, in chess the initial elements are the pieces, the board and rules of the game.
2) Trials on all level can be practically evaluated with respect to some effectiveness or efficiency criteria; also, the final proposed solution to the problem can be evaluated with respect to some such criterion.
3) The way in which the problem is solved must completely reflect the relevant laws that govern the external world for example, the solution to a marketing problem must completely reflect how consumers react to changes in styles, prices, etc.
4) Solving the problem requires only practicable amounts of computation (i.e., at a cost substantially below infinite) and the relevant information which is needed to solve the problem can be gathered by means of practicable amounts of search (i.e., at search costs substantially below infinite).
affecting the project outcome are made during the first 20% of the project's life” (Drogemuller et al., n.p.). This condition is common across the design and development fields: 85% of the total time and cost of product development in the manufacturing industry is committed in the early design stage, when only 5% of the total time and cost has been expended (Roth 1999: n.p.). The knowledge needed to guide these decisions is distributed among the different designers and makers, and the more information available earlier the better, as its impact is greater. However, this exploratory process is limited because, with few exceptions, it takes place within a single design domain: “the consensus of the industry representatives was that current modes of operation at early sketch design stage resulted in each of the architect; structural engineer; mechanical engineer; etc tending to optimise within their own specialisation” (Crawford 2004: 2) Typically, therefore, “architects and engineers operate virtually discrete processes in designing the same building” (Howrie 1995: 9).

SUMMARY
This chapter has established some of the key differences between architects and engineers. The design approaches, drivers and sensibilities of architects and engineers have been found to diverge around many factors. Representation, design process, education and materiality have served as examples that illustrate this general theme. These divergences, within the broader context of specialisation and the understanding of specialists per se in the current professional environment, have contributed to the distinct bodies of knowledge, skill and education possessed by each discipline that reinforce the situation.

Surprisingly, despite the historic and established relationship that architects and engineers share, centred on the common task of designing buildings, examination of the different roles and attitudes that each can assume when working with the other – from master to servant as a spectrum – has suggested that, with some notable exceptions, the idea exists that the activities of one discipline actively limit that of the other. As made clear by Arup’s taxonomy, when interaction between the two professions is of a poor quality it is design that loses out.

Lastly, I have introduced the idea that emerging pressures (for example time and budget limitations, design complication) are encouraging architects and engineers to engage in more integrated design exploration. I have described the early design stage as being characterised by quick investigation amidst incomplete data. This is the period during which many of the most significant decisions about building design are made, and where the costs of flexibility are least. I have also noted that 3D digital tools, connected to and supporting processes of collaboration, are seen to underpin architect engineer interaction during this time. The potential of these tools, however, has not yet been fully utilised.

While useful in establishing the underlying differences between the disciplines of architecture and engineering, and in describing the divisional nature of their interaction, the design literature drawn upon within this chapter focuses on architect/engineer conflict and therefore represents somewhat
a dead-end. In particular, when citing collaboration as a process by which architects and engineers can explore design in a more integrated manner, the term collaboration is left relatively undefined. To pursue this term, and to clarify its association with positive and productive solutions, the following chapter will move outside the traditional disciplinary literature and explore organization theory as a way of understanding the general concept of collaboration.
Chapter Four: Unravelling Interdependency

4.1 COOPERATION, COORDINATION AND COLLABORATION
4.2 DEFINING INTERDEPENDENCY
4.3 FOUR SITES OF INTERSECTION
4.4 INTERDEPENDENCY IN DESIGN
INTRODUCTION

The previous chapter set out an understanding of architects and engineers as designers who interact over shared areas of common concern (whether the driving force be desire or begrudging necessity), but who understand things differently. It was suggested that the predominantly sequential and divisional nature of their interaction is now being challenged by pressures of time, budget and design complication, as well as aspiration, and that a potential solution is for the two disciplines to interact during the early design phase. Within the literature (e.g., Achten 2002, Burry et al. 2004, Salvadori 1959, Kolarevic 2005), working across the architect-engineer boundary during this phase of design was linked to the idea of collaboration.

In practice however, the term collaboration is a slightly tarnished coin, used frequently within design literature to describe any situation in which parties work together. As Ove Arup has noted, “after all, we all agree on that [collaboration]..... But talking about it doesn’t seem to have had much effect” (Arup 1972, in Ritchie 2001: 68). Making practice-based generalisation difficult is that fact that such collaborative interactions are always, as described in the previous chapter, intertwined with the preciousness of personalities, prejudices and roles. Amongst the very few authors, Kvan in particular, who have addressed the issue of collaboration in considerable depth from within a design research context, collaboration has been linked to the idea of parties intensely working together, observing and understanding every move and intention of the other (Kvan 1999: 40, my emphasis). While providing valuable source material, these characterisations of collaboration are at one extreme too loose and at the other overly constrained, and suggest relationships different to that which I wish to propose.

Within this chapter I examine the concept of collaboration to clarify my take on the concept of interdependency. I will propose that interdependency is a condition of mutual dependence and shared problem solving. As I will discuss, collaboration is viewed as distinct from other closely related processes for managing difference across boundaries because it is characterised by interdependency, and in fact is identified as a response specific to wider conditions of interdependency.

This chapter seeks to shed light on three questions that probe the concept of interdependency further:

- How is collaboration different to cooperation and coordination?
- Why might interdependency be recognized as a beneficial mode of early design interaction between architects and engineers?
- What are the methods that support interdependency within an architect engineer design context?

To gain a perspective from outside the domain of architecture and engineering, I initialise this concept through reference to selected literature from the field of organisation theory. Collaboration
engages the attention of organisation theory for many reasons similar to those discussed in relation
to architect-engineer interaction – both fields face problems of communication and facilitating
harmonious productive work across boundaries, yet as a body of literature it has remained
relatively untapped by design theory. Viewed from this perspective, organisation theory may have
applicability in defining the characteristics of collaboration, and therefore of interdependency,
and may reveal aspects of interdependent working that are difficult to extract directly from design
literature.

From this literature, I identify and develop four themes significant to interdependent working.
These are differing perceptions, joint and creative problem solving, communication and trust.
Returning to design literature and further developing these themes within a design context, I then
suggest that the concept of collaboration as understood by organisation theory can correlate to
the process of enabling greater architect engineer interdependency.
4.1 COOPERATION, COORDINATION AND COLLABORATION

When Achten uses the phrase ‘collaborative design processes’ to describe how architects and engineers need to work together at the early stages of design (Achten 2002: 1), or Ritchie states that “it is vital that all who are to collaborate on the design of a project come together at the beginning” (Ritchie 2001: 69), we can assume that they have deliberately chosen the word collaboration over alternatives like coordination or cooperation. Often, however, these terms are used interchangeably when describing work undertaken by more than one party.

Cooperation (Fig.37), coordination (Fig.38) and collaboration (Fig.39) each describe a way of working together to achieve a particular goal. More specifically, each term describes a process for managing difference and dependency across boundaries. Whilst within the literature there is some variance in the labels, it is recognized that the three terms are differentiated by the level of interdependence each requires. While the aim of each approach is common – people look to cooperate, coordinate or collaborate when they want to achieve more than they can alone – the mechanisms are not (Fig.40).

Gray defines collaboration as a process “through which parties who see different aspects of a problem can constructively explore their differences and search for solutions that go beyond their own limited vision of what is possible” (Gray 1989: 5). This definition has proven extremely useful in developing possible models for engineer architect interdependency in this thesis, and will form a core referent. As I am using the concept of collaboration as a means for understanding interdependency, Gray’s definition provides the basis for my definition for interdependency.
Some insight into the mechanisms by which collaboration differs from cooperation and coordination can be gained from Mattessich and Monsey (1992: 39), who differentiate the three related but subtly different approaches as follows:

*Cooperation* is characterized 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 characterized 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 bring previously separated organisations into a new structure with full commitment to a common mission. Such relationships require... well defined communication channels operating on many levels. Authority is determined by the collaborative structure. Risk is much greater... and the products are shared.

Accepting this definition implies that cooperation, coordination and collaboration can be understood as an incremental set of processes for managing difference between parties, operating between independence at one extreme and interdependence at the other. *Cooperation*, whereby the parties work independently, is understood as a loose association, where information is communicated only on an as needed basis. *Coordination* facilitates dependant working. It is concerned with the management and formalisation of information exchange through defined communication channels. A number of prerequisites need to be well established to enable this to take place; clear communication, the coordination of information and the division and assignment of roles and responsibilities all act to avoid gaps in the fulfilment of a shared objective. Handy provides a concise example - “a car is equipped with an accelerator, brake and clutch. Operate them all simultaneously and to their limit, you will generate a lot of noise but no movement. Coordinate them and manage their interactions, and you progress” (Handy 1976: 212). While coordination propels the car, it offers no guarantee that it is pointed in the right direction.

*Collaborative* is characterised by shared, creative, problem solving processes. It involves interdependent working, features richer, ‘multi level’ communication channels and requires a high...
level of mutual trust (Mattessich & Monsey 1992: 39). Though difficult to achieve, the potential rewards of collaboration are such that “many organisations, in fact, now believe that the ability to get certain results can only happen through joint service efforts” (Mattessich & Monsey 1992: 6, their emphasis). Westbury echoes this statement within an architectural and engineering context, observing that “it is only out of the interplay between the two that the solution will come” (Westbury, in Castle 2002: 68). The following table (Fig.40), adapted from Handy, further clarifies collaboration by contrasting it with a bargaining approach to inter-departmental interaction.

<table>
<thead>
<tr>
<th>A bargaining approach</th>
<th>Activities</th>
<th>A collaborative approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>With regard to respective goals and orientation to decision making, each department emphasised the requirements of its own particular task, rather than the combined task.</td>
<td>Goals and orientation to decision-making</td>
<td>Each department stressed common goals whenever possible and otherwise sought to balance goals. Each party perceived the potentials for inter-departmental conflict but nevertheless stressed the importance of super-ordinate goals and the benefits of collaboration.</td>
</tr>
<tr>
<td>With respect to the strategic question of information exchange, each department minimized the other's problems or tended to ignore such considerations as it did recognize; and attempted to minimize or distort certain kinds of information communicated.</td>
<td>Information handling</td>
<td>Each department sought to understand the other's problems and to give consideration to problems of immediate concern to the other; and endeavoured to provide the other with full, timely and accurate information relevant to the joint decisions</td>
</tr>
<tr>
<td>Each department sought to gain maximum freedom for itself and to limit the degrees of freedom for the other by: emphasizing jurisdictional rules; attempting to restrict interaction patterns. Inter-departmental interactions were experienced as punishing by both sides. Contacts were limited to a few formal channels, and behaviour within these channels circumscribed by a rigid rule structure.</td>
<td>Freedom of movement</td>
<td>Each department explored ways it could increase its freedom of movement toward its goals with the following behaviour: accepting informal procedures which facilitated the task; structuring relatively open interaction patterns, searching for solutions rather than applying pressure tactics. Relations were characterized by mutual support.</td>
</tr>
<tr>
<td>Each department developed attitudes in support of the above bargaining strategy and tactics.</td>
<td>Attitudes</td>
<td>Each department adopted positive inclusive and trusting attitudes regarding the other.</td>
</tr>
</tbody>
</table>

Figure 40. Bargaining Vs Collaborative approaches (Image source: Handy 1976, adapted)
4.2 SITUATING COLLABORATION

Collaboration has been considered by organisation theory for many of the same reasons that it is of interest to my thesis. The problems faced by organisations are recognized by management theorists as becoming more complex, and “our current problem solving models frequently position participants as adversaries, pit them against one another, and leave them to operate with an incomplete appreciation of the problem and a restricted vision of what is possible” (Gray 1989: 10). The interests of businesses, communities and governments intersect on an everyday basis, either converging or colliding, and present problems that are unresponsive to conventional methods of problem resolution (Gray 1996: 57).

In outlining the characteristics of problems that organisation theory suggests require collaboration, Gray (1989: 10) lists the following:

- The problems are ill-defined, or there is disagreement about how they should be defined
- Several stakeholders have a vested interest in the problems and are interdependent
- These stakeholders are not necessarily identified a priori or organized in any systematic way
- There may be a disparity of power and/or resources for dealing with the problems among the stakeholders
- Stakeholders may have different levels of expertise and different access to information about the problems.
- The problems are often characterized by technical complexity and scientific uncertainty
- Differing perspectives on the problems often lead to adversarial relationships among the stakeholders
- Incremental or unilateral efforts to deal with the problems typically produce less than satisfactory solutions
- Existing processes for addressing the problems have proved insufficient and may even exacerbate them

Many of the items on this list closely approximate the characteristics of the early design phase and architect engineer interaction that I have identified within Chapter Three. As specialised disciplines, architects and engineers bring differing perceptions and expertise to a common, often complicated problem. Interaction between the two disciplines is organised it is often adversarial, and the sequential nature of the interaction is recognized as exacerbating the problem.

4.3 FACTORS SIGNIFICANT TO INTERDEPENDENCY

Organisations make use of collaboration as a “strategically chosen process” (Himmelman 1996: 19). In
successful collaborations, “new solutions emerge that no single party could have envisioned or enacted” (Gray 1989: 16). However, it is recognized that collaboration, and therefore interdependency, is difficult to achieve, carries potentially high costs difficult to estimate in advance (Stern 2000: 23), and challenges existing processes for interaction because it is “accomplished laterally without the hierarchical authority to which most managers are accustomed” (Gray 1989: 9 – Kennon 2006 & Salomon 2006 echo this claim within an architectural design context).

It is possible to highlight and examine many factors that impact upon and characterise collaboration. Mattessich and Monsey, for instance, identify twenty factors which influence collaboration between organisations, which they compile into six key categories: Environment, Membership, Process/Structure, Communications, Purpose and Resources (1992: 7). In developing my conceptual framework, which is concerned with understanding how 3D digital tools might enable architects and engineers to engage in more interdependent early design exploration, I will focus on four key themes which have emerged as common across the literature and my experience within Arup Melbourne. As this framework is explored over the course of this chapter and the next, I will group these themes, which I term sites of intersection, under the labels differing perceptions, shared and creative problem solving, communication and trust (Fig.41).
4.3.1 FIRST SITE OF INTERSECTION: DIFFERING PERCEPTIONS

When information is shared incompletely, differing perceptions can arise. As a project progresses, there will be different amounts and types of information generated at different phases, either from a common source or distributed sources. The ‘channelling’, or filtering, of information to particular parties and not others, or to particular sets of information that are less than the total available, leads to multiple perceptions (March & Simon 1958: 127). The greater the number of independent information sources, the greater the potential differentiation of perceptions (March & Simon 1958: 127). Ad Reinhardt’s ‘wineglasses’ (Fig. 43), in which the artist mimics the styles by which well-known modern artists might have depicted the same wine glass, graphically illustrates this condition. Different artistic styles include and exclude certain sets of information in their differing descriptions of the same glass.

Other factors that impact upon perception include the formal techniques by which information is communicated, which may or may not provide for wide communication (March & Simon 1958: 127). The extent to which the parties are connected – either spatially, by the type of work or by the type of employees (March & Simon 1958: 127) – is also important, as are the goals of the interacting parties. While it has been assumed that organisational goals are very similar –to maximize profit –the less homogenous the interacting parties the less a commonality of goals can be assumed (March & Simon 1958: 125). For example, Coxe et al describe three types of architectural firms: brains (expertise) firms, grey hair (experience) firms
and procedure (idea) firms (1987: A-1). The rewards sought by each firm, and the means by which they achieve them, differ. ‘Grey hair firms’ specialize “in producing a relatively standard product over and over again…this firm seeks high monetary rewards, but achieves them by maximizing volume”. Within ‘brains firms’, rewards “relate to security for many in the firm – increase in salaries, increase in benefits, share in profits, and growth to ownership”. For ‘procedure firms’, while not considering themselves successful until they make money, “the essential reward for this firm is, simply put, fame” (1987: A-7).

4.3.2 SECOND SITE OF INTERSECTION: SHARED AND CREATIVE PROBLEM SOLVING

Collaboration involves interaction between parties who often bring differing perceptions to a common problem, and must therefore address issues of negotiation: “when collaboration occurs, the various stakeholders bring their idiosyncratic perceptions of the problem to the negotiations” (Gray 1989: 14). Recognizing that interaction is only of value because differences occur, Gray distinguishes between differing and opposing interpretations, observing that a common assumption is that different interpretations are opposing interpretations whereas in actual fact “without differing interests, the range of possible exchanges between parties would be nonexistent” (Gray 1989: 14).

When interpretations differ, negotiation across the boundary can be understood through the concept of boundary objects. Boundary objects were introduced by Star and Griesemer (Star & Griesemer 1989) to describe objects (physical or virtual) that act as translation devices between different social worlds.
Meanings are not necessarily shared across borders and need to be reconciled, as objects and methods mean different things to different people (Star & Griesemer 1989: 393) - for example, “the depiction of a welded joint may stand for part of the support structure to the designer and for labour extended to those in the shop” (Henderson 1998: 54). While recognising that consensus is not necessary for the successful conduct of work, boundary objects provide an understanding of how interaction between architects and engineers can establish a common representational ground.

Because boundary objects reside at the interface of different interacting parties they perform a brokering role in situations where there is a symmetry of ignorance (Fischer 2000: 529). These are situations where no single party in the collaboration holds all the knowledge – often the case when specialists need to communicate discipline specific knowledge across domain boundaries.

4.3.3 THIRD SITE OF INTERSECTION: COMMUNICATION

For collaboration to occur, information must be well communicated across boundaries via information or communication ‘channels’ (Handy 1976: 233). Factors that can lead to poor communication include perceptual bias by the receiver, omission or distortion by the sender, lack of trust, information overload, information secretion and a lack of clarity (Handy 1976: 355).

In cooperative processes, each organisation functions separately and conveys information only as needed, without any level of formalization. In coordinative processes dependent working is achieved when “communication rules are established and definite channels are created for interaction” (Mattessich & Monsey 1992: 40). In collaborative processes, joint strategies are developed that rely on “many ‘levels’ of communication” (1992: 40). These additional ‘levels’ might be the establishment of informal communication channels, which supplement the established formal ones and provide information beyond the base level of information required but necessary to complete the work, and/or the adoption of a ‘language’
that increases the efficiency of communication by “making it possible to communicate large amounts of information with relatively few symbols” (March & Simon 1958: 162). For any communication, an important aspect is efficiency: “the greater the efficiency of communication… the greater the tolerance for interdependence” (March & Simon 1958: 162).

4.3.4 FOURTH SITE OF INTERSECTION: TRUST

Interdependent working, in which tasks that cannot be achieved by any one party, can “allow many other people to get into positions where they can, if they choose, injure what we care about, since those are the same positions that they must be in order to help us take care of what we care about” (Baier 1986: 236). Risk and interdependence are necessary conditions for trust (Rousseau et al 1998: 395). Within organisation theory, trust is “regarded as an important coordination mechanism” (Lane 2001: 1) and particularly as a means for managing risk, uncertainty and reducing apparent complexity. According to Meyerson et al, trust facilitates a “collective perception that is capable of managing issues of vulnerability, uncertainty, risk and expectations” (Meyerson et al 1996: 167); in situations where trust is lacking, the reliance on checking mechanisms and tested, formalized procedures increases: “if we do not trust somebody, we are careful to screen the information” (Handy 19: 355).

While “to date, we have had no universally accepted scholarly definition of trust” (Rousseau et al 1998: 394), the dominant theory has until recently been the ‘rational choice’ model. This is an essentially calculative conception (Casson & Cox 2002: 178). Within this model, social interaction is seen as primarily governed by self-interest, with each party seeking to maximize individual gains and to protect themselves from the opportunistic behaviour of others. Trust is viewed as a function of predictability (Lewicki & Bunker 1996: 121) - the confidence that another party will act in a certain way because they have acted that way previously – and is built up through repeated interaction (Powell 1996: 60).

The rational framework has led to a “restricted focus on the efforts of self interested individuals to achieve optimum outcomes in interactions with particular others” (Tyler & Kramer 1996: 2), and is challenged by the movement from hierarchical to lateral modes of interaction within and between organisations. Of particular relevance to this research is the problem of building trust in temporary groups. Organisation theory recognizes that “organisations are moving away from formal hierarchical structures to more flexible and temporary groupings around particular projects” (Tyler & Kramer 1996: 8). Similarly, within the construction industry, “projects are usually temporary alliances of autonomous partners” (Samuelsson 2003: 225), and alliances are traditionally organised around the concept of the
lowest tender (Cornick & Mather 1999: 108). “In many respects, such groups constitute an interesting organisational analogue of a ‘one night stand’. They have a finite life span, form around a shared and relatively clear goal or purpose, and their success depends on a tight and coordinated coupling of activity” (Meyerson et al 1996: 167).

Within these groups, there is not the time to develop trust in a traditional sense. “Temporary groups often work on tasks with a high degree of complexity, yet they lack the formal structures that facilitate coordination… they depend on an elaborate body of collective knowledge and diverse skills, yet individuals have little time to sort out who knows precisely what” (Meyerson et al 1996: 167). An alternate means has been suggested by Hardy et al. who identify a communicative foundation for trust: “in an inter-organisational relationship, trust grows out of a communication process in which shared meanings develop to provide the necessary foundation for non-opportunistic behaviour. Accordingly, trust can be conceptualised as a communicative, sense making process that bridges disparate groups” (Hardy et al 1998: 69)

SYNOPSIS

This section has examined collaboration, as it is understood within organisation theory, to clarify the concept of interdependency. Interdependency is the factor that differentiates collaboration from other closely related processes for managing difference across boundaries. Drawing on Gray, interdependency has been defined as a process through which parties who see different aspects of a problem can constructively explore their differences and search for solutions that go beyond their own limited vision of what is possible (Gray 1989: 5). In terms of design interaction between architects and engineers, cooperation can be assumed to be inherent in every architect engineer interaction. Coordination develops clear communication channels that support dependencies between parties. Collaboration can be differentiated from cooperation and coordination because it features interdependent working, characteristics of which include mutual responsibility, mutual dependence and mutual trust. Interdependent working is accomplished laterally and via shared approaches to problem solving.

Within organisation theory collaboration is associated with problems that are ill-defined and complicated, and where differing perspectives can lead to adversarial relationships. These characteristics echo those I have detailed as affecting the architect engineer relationship in Chapter Two. Four sites of intersection which offer useful concepts for analysing the architect engineer relationship have been identified. These are Differing perceptions, Joint and creative problem solving, Communication and Trust.

4.4 INTERDEPENDENCY IN DESIGN

It is possible to further develop the concept of interdependency, again via the concept of collaboration, through design literature. Returning to this literature provides a means to establish a correlation between organisation theory and design perspectives on interdependency. Again, interdependency will be examined through the concept of collaboration. These synergies and
overlaps can then be placed alongside the models and concepts that I have established of architect engineer interaction.

Examining interdependency within the context of architect engineer interaction will extend the four concepts outlined above. In particular, three questions that arise in considering design literature are important in further clarifying the understanding of interdependency within this thesis:

- Collaboration has been described as a strategically chosen process: is it appropriate for early design exploration?
- By what methods can interdependent working take place between designers?
- What makes interdependency difficult within the design world?

Kvan, in examining distributed design contexts and 'distal' design communication, describes a strong connection between collaboration and the creative aspects of design. As opposed to cooperation, collaboration involves joint and creative problem solving: “the important distinction between the two words is in the creative aspect of working together” (Kvan 1997: 6). Collaboration means “digging into issues to find innovative possibilities… being open and exploratory. It implies a deep level of trust and acceptance” (1997: 6), and “requires more than machinery and systems to occur” (1997: 6).

This creative aspect of collaboration poses design problems, and Kvan identifies collaboration as “a far more demanding activity, more difficult to establish and sustain, than simply completing a project as a team. I suspect that we collaborate far less often than we pretend to” (Kvan 1999: 39). It is time-intensive and “suited to very particular problems” (1999: 102), and is therefore not an appropriate means of accomplishing many design tasks. Significantly, Kvan states that “collaboration occurs as negotiation and evaluation” (1999: 40); this idea will be closely examined in the following chapter. For collaboration to occur, in Kvan’s evaluation, requires design processes be closely, rather than loosely, coupled, a situation that Kvan notes is rare. A close coupling is one in which “the participants work intensely with one another, observing and understanding each others moves, the reasoning behind them and the intentions” (1999: 41). This correlates with Gray’s description of collaboration as a lateral activity. Within loose coupled processes, designers “work together for moments, then divide up and go their separate ways” (1999: 44), returning to haggle over the results.

In describing the difference between three types of cross-disciplinary working (intradisiplinary, interdisciplinary and transdisciplinary), Zeisel identifies some methods by which design might be thought of as ‘close-coupled’ (Kvan 1999: 41). While using terminology linked to interdisciplinary studies, his schema corresponds to that of Mattessich and Monsey. Zeisel describes intradisciplinarity as one discipline acting as a sub consultant to another, while interdisciplinarity is the division of a problem into separate parts, coordinated and carried out in parallel (Zeisel 1981: 53). Closely approximating Mattessich and Monsey’s definition of coordination from organisation theory, Zeisel describes this mode of working as one in which “responsibility for each part remains separate, but
team members have joint responsibility for the quality of the links” (1981: 53).

Transdisciplinary procedures involve the development of new procedures and decision making processes that “neither wholly reflect any one discipline nor join different disciplines” (1981: 53). Parties share in decision making and responsibility, and develop clear processes of using information in design. New types of shared methods, problem definitions and tasks are developed, which increase the “number and types of tools [participants] can use separately” (1981: 53) and extend the skill base of all involved. This procedure is thus “the most productive form of practice” (1981: 54). Zeisel notes that all these forms of cooperative practice share possible difficulties in communication, power structures and responsibility, and that while working cooperatively there is always “a natural tendency [for the disciplines] to retreat into their own disciplinary shells and say “your test results are not relevant for me”” (1981: 56).

Some insight into the issues that ‘new shared methods’ must address is provided by Achten, who specifically addresses collaboration in engineering design. Similarly to Kvan, he recognises that “technology alone will not be enough to make this change happen” (Achten 2002: 9) – engineering designers need problem solving strategies that can deal with “incomplete, inconclusive, and changing information in the design process” (2002: 10) and their work, undertaken in these ill-defined environments, “needs to be differently appraised by other design participants” (2002: 10). For the engineer, being involved in collaborative design processes demands new design support tools and design processes, and on the provision of “insight in the design goals and problems of other participants” (2002: 4).

Achten makes several distinctions similar to those made previously. He describes cooperative design as a process involving problem decomposition, task assignment and solution integration, in which participants are strictly bound to their parts of the problem: “participants get such parts to solve and later integrate in partial solutions that are again integrated in a whole design” (2002: 4). Collaboration, in contrast, involves participants contributing to the design work and design problems of others (2002: 4). In providing a tentative definition of collaboration, he states that “collaborative design is a process in which the participants work together in a meaningful way, not just working together efficiently, but stimulating each other to contribute to the design task. They act towards mutual understanding and maximising outcomes that satisfy not only their own respective goals, but also those of other perspectives” (2002: 7). The idea that collaborators might stimulate each other corresponds to Zeisel’s idea that collaboration can involve increasing the potential of the other party.

Practice examples of new design processes that allow architects and engineers to work together in a meaningful way are described by Kloft (2006), who uses the term collaboration in an overarching manner to describe three very different interactions. These interactions clearly demonstrate the impact that cooperation, coordination and collaboration can have on the architect engineer relationship. The first type of interaction, which Kloft terms ‘forming a shape’, describes the traditional approach to the architect engineer relationship. As I have described in the previous chapter, this relationship involves the engineer realizing a structural design after the architectural
form generation (a wholly independent process) is substantially complete. Kloft illustrates this approach through Bernhard Franken’s BMW Bubble project (1999), in which a digital model is generated according to architectural concerns and is provided to the engineer as an untouchable ‘master geometry’. The engineer’s aim is to “support the formal idea… [by] materializing the generated shape and blowing up the digital model to its structural proportion” (Kloft 2006: 85).

The second type of interaction, ‘forming a form’, describes a process whereby a form designed by the architect is open to change through interaction with the engineer, although this interaction does not take place during the form generation process: “structural behaviour [is] allowed to have an influence on the final geometry” (2006: 86). Again, this type of interaction is common between architects and engineers. The process is iterative and formal and structural development and optimisation take place via close collaboration between the architect and engineer.

The third kind of interaction, ‘generating a form’, describes a process of collaborative form generation, with architect and engineer working together from the very beginning of the design process to integrate structural design and form generation. Kloft states that “the challenge of generating a form in this manner is to combine the architectural intention and formal design freedom with engineering creativity in regard to the rules of stress flow” (2006: 90). This type of interaction is an example of new solutions emerging that “no single party could have envisioned or enacted” (Gray 1989: 16).

Ritchie, similarly reflecting on practice-based experiences of collaboration, locates mutual respect, trust and the commonality of shared objectives as fundamental for collaboration. He locates design quality and similar values as examples of these objectives (Ritchie 2001: 68), echoing March & Simon’s discussion of perceptions within organisation theory (March & Simon 1958: 127). Ritchie states that shared objectives often need to be actively constructed: “mutual education and reorientation are necessary when a job comes to us with another consultant already attached to it… and there is a heat period necessary to melt the engineering and architectural boundaries” (Ritchie 2001: 67). Cumming also raises this issue, noting that in architect-engineer interactions “people from different cultures, who may have never worked together before, are brought together and expected to quickly bridge striking cultural differences and become productive with one another” (Cumming 2002: 270).

Ritchie presents ‘ten commandments for collaboration’, most of which reinforce the ideas previously associated with collaboration: that all participants are equal, ideas are shared by the team and not claimable by any one party afterwards, and that the process must include time together and time alone. Ritchie’s first commandment, that there has to be a moral commitment to the concept of collaboration (Ritchie 2001: 68), puts front and centre the idea that, as Kvan has also noted, collaboration is a demanding activity: “when we collaborate, some kind of friction always arises, whether it is over the money, the design or the morals…” (2001: 68). While the morals that Ritchie refers to are slightly unclear, clarification might be found in Ove Arup’s requirement for integrated design that all members of the team “want to help to produce good architecture, architecture in depth, so to speak” (Arup 1985: 9).
SUMMARY
Within this chapter, I have reviewed organisation and design theory to define and understand aspects of interdependency that may then be productively brought back to inform the architect engineer relationship. I have initially defined interdependency through reference to Gray’s definition of collaboration: as a process through which parties who see different aspects of a problem can constructively explore their differences and search for solutions that go beyond their own limited vision of what is possible. Building upon this definition, within this thesis I will understand interdependency as follows:

Interdependency is a productive form of practice enabled by mutual and lateral dependence. Interdependent parties use problem solving processes that meet not only their own respective goals, but also those of others, by constructively engaging difference across their boundaries to actively search for solutions that go beyond the limits of singular domains.

I have identified four sites of intersection significant to interdependency; these are differing perceptions, establishing common meanings, communication and trust, from the perspective of organisation theory. I have then reviewed design literature, which suggests that there is a correlation between these sites and issues that affect architect engineer interaction, and that interdependent working is suited to the early design exploration phase.

To summarise the four themes:

Differing perceptions: Different disciplines bring different goals, interpretations, levels of expertise and access to information to a common problem. When the information they share about that problem is too limited, or comes from many independent sources, differences of perception can arise.

Shared and Creative Problem Solving: Without differences between the parties, the range of possible exchanges would be nonexistent. Within interdependent processes, establishing common meanings involves negotiation, which is a creative exercise. Shared and creative problem solving occurs through negotiation and evaluation, which requires close coupled design processes and new tools and approaches that are suitable to the early design phase. The design results neither belong to one discipline nor join different disciplines.

Communication: Efficiency in communication facilitates interdependency. In interdependent processes, communication needs to occur at multiple ‘levels’ to provide information supplementary to the minimum. It has been suggested that this supplementary information might include insight into design goals of others, the reasoning behind them and the intentions of the designer.

Trust: The issue of trust recognises that technology alone is not enough to support interdependency. Trust facilitates interdependency by managing and reducing apparent risk. Without trust,
information is screened and the goals of other perspectives are ignored. In examining the phenomenon of temporary groups, it has been suggested that there may be a communicative foundation for trust.

Additionally, I have identified that architects and engineers may require new types of shared methods and tasks for shared problem solving. These methods need to be compatible with the incomplete, inconclusive, and changing information characteristic of the early design process. Potentially, this development of new types of methods and tasks for problem solving requires a mutual education and reorientation for architects and engineers. To understand why new methodologies and forms of communication are now being called for, the following chapter will extend these themes in two ways: by investigating their impact upon the use of digital tools by architects and engineers, and through reference to interview data that describes how problems are known and experienced by practice.
Chapter Five: Intersections & Interdependency between Architects and Engineers

5.1 DIFFERING PERCEPTIONS:
  PROBLEMS OF INTERPRETATION BETWEEN CAD AND ANALYTIC TOOLS

5.2 SHARED AND CREATIVE PROBLEM SOLVING:
  PERFORMANCE-BASED DESIGN

5.3 COMMUNICATION:
  BACK-PROPAGATING DESIGN INFORMATION

5.4 TRUST:
  REPRESENTATION & EXPLICATION
INTRODUCTION

From the design and management literature referenced in the previous chapter, one can draw some conclusions. Cooperation, coordination and collaboration have been understood as related processes for managing difference across boundaries. They are nested processes, thus the characteristics of cooperation are inherent in coordination, and those of coordination inherent in collaboration. Collaboration, as opposed to cooperation and coordination, has potential to support architect engineer interaction during the early design stage because it involves interdependent working. The concept of collaboration developed within organisation theory has therefore provided an effective means for understanding interdependency.

Four key themes were identified in the previous chapter: differing perceptions, shared problem solving, communication and trust. While these themes have not yet been examined closely within the contexts of 3D digital design tools and their potential for architect engineer interaction, some broader aspects of the problem have been revealed. It has been suggested that architects and engineers require new types of shared methods and tasks for shared problem solving. Further, these new methods need to be compatible with incomplete, inconclusive, and changing information in the design process. Potentially, this requires a mutual education and reorientation for architects and engineers.

To further extend these themes, and to shift from the literature to a more hands on discussion developed out of practice, in this chapter I investigate several problems related to early design exploration that impact upon the use of digital tools by architects and engineers. I will connect these problems to the way engineers understand and experience them in practice by presenting information gained through an interview process that has captured the experiences and views of nine Arup engineers from three offices. This chapter concludes my development of the conceptual framework, and ends by posing four questions which will be explored through practice-based projects in the following chapter.

1 A detailed description can be found in the Table of Interviews, and in Chapter Two 'Approach to Research and Method'.
5.1 DIFFERING PERCEPTIONS: PROBLEMS OF INTERPRETATION BETWEEN CAD AND ANALYTIC SOFTWARE

Within the previous chapter, the problem of differing perceptions was identified as a limiting factor for interdependent working. Differing parties bring different goals, interpretations, levels of expertise and access to information to a common problem. For architects and engineers, differing perceptions are manifest in the different representations used by each discipline, the different deployments to which those representations are put, and in the problems associated with transferring representations of design information between those different deployments.

The problem of linking design representations to different deployments is a difficult and longstanding one. Evidence of the complexities involved is provided by Brown (2000) who, in exploring the history of British and American engineering drawings, relates a story of the British government’s 1940 decision to purchase 60 frigates from American yards. This involved the production of complete plans, for what would become the Liberty freighter (Fig. 47), which were drafted in Britain and sent to American shipyards. Despite American experience in ship building, the ‘universal language’ of engineering drawing and the same tools being used in each country, "much to the commissioners’ surprise and dismay, their ship and engine plans proved essentially meaningless to managers and workers at the American yards. The entire set had to be redrafted, and hundreds of additional drawings were needed before work could begin" (Brown 2000: 195). It was discovered that British practice was to make only about 30 per cent as many drawings as was customary in American yards, a result of procurement routes that had diverged due to different values, institutions, social relations and the American system of manufacture². Despite the use of a precise and international language, being engineering drawing, the communication of design information between British designers and American shipyards was not successful because it did not take into account the particulars of the deployment.

Figure 47. Liberty freighters (Image source: http://www.merchantnavyofficers.com/liberty3.html)

² Brown describes how the ‘American system’ of manufacture meant that America only began using detailed drawings in the 1870s, approximately 40 years after the British. Instead, the production of standardized models, made possible for the first time by the American system, relied on precision and the subdivision of labor, and therefore the use of jigs and templates to describe design, rather than drawings.
5.1.1 TRANSLATION AND INTERPRETATION

Similar problems arise in architect engineer communication when information is moved across the boundary, and is particularly the case when design information is transferred between the CAD tools used by the architect and analytic tools, employed by the engineer. Moving information between these different representations is made difficult by several factors, including the semantic distance between many of the software tools and the relative levels of geometric precision that CAD and analytic tools require. The exchange of information still remains an inefficient and time intensive process.

One aspect of this problem is translation. Translation is central to the design process, whether in the form of translations between software tools (cad-analysis-manufacture), translations between design stages (conceptual-detailed-documentation), or the translation between drawing and building. As Eastman has noted, “design involves creating information in one representation, then transferring it to others, until the composition satisfies diverse criteria that are evaluated in the different representations. In both manual and CAD based design, each representation of an element is defined and managed separately by the designer, requiring significant effort in multiple translation and coordination. Translation and coordination is a major aspect of design” (Eastman 1991: 18). The processes of translation include both the sequential development from sketch to documentation of ever more precise design descriptions, which have been characterized as longitudinal translations (Fig.48), and a more iterative set of latitudinal translations (Fig.49) between the different design descriptions which inform the growing precision of the design.

This second set of translations occurs between software programs that are particular to each discipline. As observed by Akin et al, “where design tools once were of more or less common nature in different design areas (pen and paper, physical models), various disciplines have developed specialised software to suite their specific needs” (Akin et al 2003: 246). Within this specialised software, each discipline employs different digital representations to facilitate its work. A good example of the type of representations that may be generated for any single design object is provided by Cornick (1996: 98), who describes a common drawn object, in this case a beam to beam connection, as representing:

- To the architect, an architectural feature that will have a perceived visual and functional impact.
- To the structural engineer, a structural element that will transmit forces and perform as part of a larger structural system.
- To the service engineer, a physical element that may or may not support or be compatible with the installation of services.

3 I am not suggesting that this problem is limited to the transfer of information between CAD and analytic tools, as it can also affect the transfer of geometry from one CAD program to another. During informal conversation with the senior CAD technician at Arup Melbourne, he described difficulties in transferring 2D information between AutoCAD and Microstation. These difficulties were encountered during the design of Ashton Raggatt McDougall's National Museum of Australia (2001); the result of different underlying mathematical descriptions for curves within each CAD program.
To the trade contractor, a construction element that requires detailed design, and the on or off site application of skilled work, material and plant.

To the construction manager, a construction element installed by others that may or may not meet a given time and cost.

The translation of information between these specialist representations is not well supported by existing processes, as upcoming interview data will evidence. As a result geometry generated in an authoring CAD environment is often not immediately useful in other deployments, or missing information that must be added to complete that discipline’s task (Fig.50). In particular, the different informational requirements of architectural and engineering software mean that significant manual rework, or ‘cleaning’, is required before information from the CAD software can be used by the downstream analytic software. Cleaning, carried out by the engineers and CAD technicians, can involve tasks such as manually tracing over all the surfaces in a 3D model, finding every single intersection point, extracting the centrelines from 3D entities, simplifying that geometry or simply assigning different layers or other information to geometry. Often, cleaning a model can take longer than the subsequent processes of analysis.

Significantly, it can also involve tasks that adapt or interpret geometry in reference to a particular problem. For instance, in performing CFD analysis on a grilled façade, an engineer would not use the ‘real’ geometry of that façade (Fig.51) because doing so would be highly costly in calculation time within the simulation. Instead, a far more efficient approach is to substitute the grill (which contains potentially thousands of small holes) with a small number of transparent zones that match the properties of the ‘real’ grill (Fig.52). Such a model can be analysed very quickly, however it is important to note that a wide variety of possible user errors, including usage errors, physical approximation errors and discretisation errors, can lead to solutions that may be in complete variance to real flow. Results should therefore be treated with extreme caution.
5.1.2 THE EXPERIENCES AND UNDERSTANDINGS OF PRACTICE
The interview extracts within this chapter and chapter two are drawn from interviews I have conducted for the purpose of accessing the real life experiences and opinions of highly experienced practitioners within Arup. They are presented here and in the following three sections to reaffirm, and possibly contradict, aspects of the more theoretically derived ideas presented in this and the previous chapter.

The current requirement for ‘cleaning’, or the manual reinterpretation of design data, is concisely summarised by Mr. C:

\[ \text{Rhino}^\text{TM} \text{ can output something, but it's not what my program needs} \]

whilst Mr. B provides additional detail into the practicalities of the process:

\[ \text{If the architect's done the 3D model, we would probably use that, but it's never a straight drag and drop. We might have to go around and rebuild all the surfaces} \]

While addressing technical aspects of this problem are a primary concern for engineers, there is the recognition that a lack of understanding between the disciplines is a contributing factor. The concern expressed is that architects misunderstand the work of engineers because they are not exposed to crucial aspects of that process, particularly with regard to analysis:

\[ \text{The architect I worked with on a project just before coming to Melbourne was a co-located project, so we actually sat} \]
next to each other. And the response I got from him was that “I really didn’t understand what
engineers did, because all we see from you is a very simple line diagram with bold lines and some
less bold lines and some sizes of elements on them, and you wonder what on earth are we paying for,
how difficult can it really be to produce those drawings. And it’s by having worked with you guys,
and seeing what goes into that I understand that that’s just the end game, the final representation”.
But that doesn’t in any way represent the process to produce it, and the technical expertise that
produces it.
Mr. A

With clients, often it’s ‘slide A, slide B, bad, good, thanks’. It really depends on the client — but
I don’t know what goes through someone’s head when they see all that money is spent and it comes
down to two slides. But I suppose that it’s similar to the structural team - at the end of the day the
outcome is a plan, or a set of drawings, whatever the process they have used.
Mr. C

Further evidence supporting these statements can be found in section 5.2.4, and in Lam et al’s
survey of 584 architectural and engineering firms in Singapore which found that “most firms viewed
the use of simulation tools as involving extra cost and effort but with very little appreciation from the clients” (Lam
et al 1999: n.p.).

Even within projects where simulation and analysis is an integral part of the generative design
process, interaction between the architect and engineer occurs only through a very narrow interface.
As with the statements of Mr. A and Mr. C, Mr. I’s statement below reveals that representations
which promote easy communication are not necessarily the same as those that aid in the design
process:

There were a number of interfaces – one of the most important was a piece of software that
unfolded the building, and it was this unfolded version that we passed backwards and forwards with
the architect. It was very good for visualisation - with the analysis information mapped on, you
could see what elements were utilised and to what extent. It was a semi automated process that ran
from SAP that allowed for an iterated, optimisation process.

Thereafter, when there were changes, it was a tedious task of moving nodes etc. Interaction with the
architect was generally in a 2D manner. When we had workshops there were physical models there,
so there was always that aspect there. They went about it in a very architectural way, producing lots
of scaled models. Early on the models didn’t incorporate any of the structural stuff, but gradually
they incorporated our structural sizes to determine what they would look like.
In looking to the potentials of technology to resolve issues of translation and interpretation, two different approaches were described. Mr. D advocated the use of an (as yet non-existent) single, integrated tool:

*The best model is for a common tool within every discipline's desktop, a common big super-powerful tool, but with all the values defaulted. If you look at how they make movies, all the designers use the same software, Alias™, but each one of the designers will look at different things – one will look at the movement of people, the other will look at trees, etc – they all work in the same environment but each designer works with a family of parameters and knows how to optimise for that. Something similar will happen for us. The architect will define zones, say a mechanical zone, but that will be a default zone. That default will then pass to the specialist mechanical designer who will be able to tweak all the parameters. When I say the software will be the same for everyone it will be a common platform, but each specialist will have their own plug-ins.*

A concrete example that comes close to approximating Mr. D's future-based description is provided by Mr. H, who discussed the use of Digital Project™ within a live project:

*The idea was not to use 2D drawings, but to build up one single model that contained architectural information, structural information and building services information, and try to eliminate any clashes.*

*We built up the structural frame first and the architect put the curtain wall on top of our structural elements. Afterwards the contractors got in and filled up the model. There were a lot of clashes – every type – the client estimated more than a million dollars worth if they had been built on site.*

*When the contractors pick up the model, which has a central core, a megatrust and megacolumns, they pick up that small part and build a very detailed model. For example they put in all the reinforcement, all the stairwells etc, and that's where the clashes start occurring.*

*They used the model to produce a 4D model. We put attributes into the concrete elements, including concrete grain and steel ratio, so that the QS can cost it, which is happening now. After completion of the project, the model will handover to the Building Management Office. The contractor will input all the relevant information into the model or link with the model. By clicking on objects within the model they will then get the relevant information instead of retrieving it from a file or other conventional methods.*

It is interesting to reflect on Mr. H's description of the modelling process and the detection of clashes in the light of March and Simon's organisation theory-based assertion that the channelling, or limiting, of information to particular sets that are less than the total available leads to problems of multiple perception (March & Simon 1958: 127). While outside the scope of my investigations, it could be suggested that the modelling process, in which different parties picked up only small
parts of the model to detail more thoroughly, contributed to the number of clashes eventually detected.

Mr. G, who is more design oriented, takes a different view. He suggests that a fully integrated software platform might not offer the optimal solution. He states that:

I am not the best at collaborating in the sense that everyone in the world should use these tools... For instance you would have your working environment and be constantly evolving it, as I would mine. Where it is appropriate for them to interact that’s what should happen... In some ways it would be better to do them all in the one environment, but I don’t think that is really practical, because it is horses for courses.

The techniques that I started using, which were Microstation BASICTM, Excel Visual Basic for Applications™ (VBA) and then Microstation™ Visual Basic once they updated it, have also been used by people like Foster, who have done some of the most sophisticated stuff around. It is not really a friendly way of doing things, not really, however it is very good for some things. It has its place and CATIA™ has its place. Maybe we shouldn’t worry about it too much – it’s too early to be heading for one environment. It’s frustrating but perhaps that’s just the way things are, and it will probably remain that way for 20 years. It’s better than locking yourself into something like AutoCAD™, which people did far too early when it wasn’t very good and lo and behold they’ve stuck with it ever since.

5.1.3 CAD APPROACHES TO TRANSLATION AND INTERPRETATION

Several approaches to resolving problems of translation and interpretation described above are currently the focus of considerable research, investigation and review (e.g. Jaroslaw 2002, Plume & Mitchell 2005). While I will not examine these in any detail, one such area is the development of common languages, such as the neutral file formats STEP or IFC, and on the way these file formats can convey multilayered information between different parties. A second area is the management of information, for instance the development of integrated databases, to which all members of the team have controlled access, or of federated databases, where discipline specific aspects of project information are held at different locations and accessed via a standard interface.

The problem of integration is addressed with some clarity by Mahdavi, who suggests that there is a bi-polar approach (Mahdavi 1998: n.p.). Within CAD research, approaches have been either strategic, “involving the program for an all encompassing and yet highly detailed building model involving a maximal building representation”, or pragmatic, “involving the ad hoc and as-needed production of translator and mediator routines” (1998: n.p.). Claiming that neither approach has been very successful, he offers two related explanations. The first of these is that discipline specific views (typically belonging to architectural developers) may conceal the operational requirements and domain specific knowledge
associated with analytic software. This may lead the developers of large and all-encompassing building product models to treat technical programs as isolated black boxes and fail to properly understand the characteristics of those programs and their use. This observation corresponds with the above stated views of Mr. A and Mr. C, that architects may not fully appreciate the engineering workflow. Mr. C provides some insight into the potentially complicated nature of these operational requirements:

*The thing that kills you in CFD is making the computational mesh, because it’s so picky – if your mesh is bad your model crashes, and you have to go back a long way to fix these problems. You build your geometry, you build your mesh, you set up all the physics and you run it and it crashes, and the crash might be due to a geometric problem which is way back where you started. It’s not a matter of going back and making a little tweak to get it running, you have to go right back and do it all again. There are some tools out there that remove some of the control, and so improve the stability of the process: when you build a room it makes a square room and fills it with square mesh and there’s never any problem. But the drawback is when you want to model the Swiss RE building you can’t do it, because that program doesn’t have the flexibility.*

The second explanation offered by Mahdavi concerns the idea that representations can be divorced from functions, and that it is possible to construct “the representation of the building, which gives any kind of information, from any point of view” (Mahdavi 1998: n.p, his italics). Arguing against the view that a building representation, as a member of a class of real things (or being made from a collection of real things), can find corresponding referents to those real things in different special domain representations, he states that “it is difficult to defend the notion that building representations may be constructed (or shall we say discovered?) irrespective of the intentional and functional interests of the agents that do something with those representations” (1998: n.p.). My experience within Arup Melbourne coincides with this view, as does some of the interview data.

This crucial impact of use and deployment appears in a study described by Johnson et al (2004). Attempting to translate information between architectural descriptions generated in a Building Information Modeling (BIM) type 3D object oriented modeling environment, Proteus, and Finite Element Analysis (FEA) based structural analysis program, the authors describe how the process is firstly complicated by domain specific views and secondly by interpretive elements. In this case, the initial representation of geometry, or the architectural view, is achieved using objects that correspond to ‘real’ building elements (eg. 3D I-beams), which are drawn as a continuous longer element crossing a series of supports.

When translated to the FEA software, this same information needs to be represented as a series of short centreline segments each stretching from one support to the next (noded connections). In other words, the ‘real’ 3D representation of the beams and columns need to be simplified into an idealized form – in this case, noded centreline geometry. By itself, that it not an unachievable task – the current version of Bentley Structural™ can silently build a noded wire-frame model
behind a ‘real’ building model. However, to achieve a FEA representation that is ‘analysable’, the authors describe several additional and necessary operations: the structural model required ‘offsets’ (infinite strength, zero length connections) and other interpretive elements that did not appear in the architectural representation. These elements were necessary to produce an accurate and workable analysis model. Finding that the CAD and analytic representations did not correspond in a one to one manner with one another, the authors conclude that “if an architectural model itself is comprised of elements meant to correspond to actual building components… then the architectural model itself is not going to be appropriate for structural analysis” (Johnson et al 2004: 237).

5.1.4 A PRAGMATIC APPROACH
This issue is worth pausing over because it is significant to the way I have approached the use of 3D digital tools within the early design phase. I have noted that an aim of this phase of exploration is to inform a design “within a context of incomplete and imprecise data” (Akin 1998 n.p.) but, as Fekete has pointed out, “striving to model fully in 3D has become equal to the quest to achieve as detailed a representation of the finished building as possible” (Fekete 2003: 249, his emphasis). To facilitate this quest, CAD research into object oriented modeling has focused on increasing the number of symbols representing real things. As Fekete continues,

“apart from the fact that ready-to-use symbols have to be preconceived for objects not yet thought of, let alone defined, a preconception of an entire domain, i.e. architecture, becomes inevitable… a sweeping statement has been made that walls, floors, roofs, pillars, slabs, doors, windows and various installations and furniture (sum total) are sufficient artefacts to model every conceivable part of a building”
(Fekete 2003: 249).

If this were indeed true, then design exploration would not occur in a context of incomplete and imprecise data, but would be similar to using Lego™ (Mr. C has already alluded to the problems associated with using square rooms to model the Swiss RE4). Object-oriented CAD software such as Revit™ increasingly provides the designer with the means to create their own design elements, though this currently holds a particular irony. Whilst these user created objects arguably embody a higher level of the designer’s intelligence than those ‘off the shelf’, the time cost of adding attributes and properties to these objects are such that they end up carrying the least embedded intelligence.

Within this thesis, I have taken a pragmatic, in-situ (interfacing) approach. This approach looks to find meaning in actions that permit the investigation to continue – in this case, communicating between CAD and analytic softwares – and accepts that this is related to interpretations and understanding in practice. These actions will take into account knowledge of data translation,

4 The Swiss Re, also known as 30 St Mary Axe or the ‘gherkin’, was designed by Foster and Partners, London, 2000 to 2004.
design intentions and workflow that are accessible by virtue of the ‘embedded’ nature of my research investigations with the Arup Melbourne office. This approach recognises that “at some moment, by some means, the specifics of how people work become crucial to the design of working systems” (Suchman 1995: 61), and that this information can inform mechanisms for ‘getting on with the job’. Burry et al, in describing the transfer of design information into the parametric modeler CADDSTM, provide a clear example of this approach:

“In the case of CADDSTM, we resorted to an amalgam of mathematical calculation via MS Excel™ to derive the appropriate geometries, and MS Word™ via macros to convert that information into executable scripts for CADDSTM. To the computer-using aficionado, reverting to standard desktop in order to produce the executable scripts probably seems top-heavy. On the basis of ‘just getting on with the job’, such purism is interesting, but not especially relevant” (Burry et al 2001: 79)

A simple example is the transfer of geometry from the CAD software Rhino™ to the structural analysis software GSA™. Typically, a model is designed within Rhino™ using NURBS5 surfaces (Fig.53), but is exported as a Drawing Exchange Format (DXF) file which can be imported into GSA™. Often architects will provide this DXF file themselves. Modeled ‘architecturally’, as one continuous surface that splits another into two pieces, the DXF mesh that is generated does not produce the noded mesh (Fig.54) geometry along the intersection line required by GSA™. While the DXF mesh will import into GSA™ it will not solve properly, and will require remodeling within GSA™ by the structural engineer. Knowing that this geometry is required (knowledge that I have gained by working within the Arup Melbourne office) the ‘MatchMeshEdge’ command can be used within Rhino™ to provide a DXF mesh geometry that requires no cleaning (Fig.55). From very little effort in Rhino™ it is possible to save hours of manual rework later on.

More complicated examples of a pragmatic approach are used by engineers within Arup, with two interviewees describing the development of their own interfaces between CAD and analytic software. Mr. C uses scripting to directly manipulate the files imported and exported by CAD and analytic software:

I’ll munge6 that file and along the way feed in some parameters. Otherwise you have to do it by hand or with an Excel™ spreadsheet, which takes forever. I often use them as a throwaway tool. Say you’ve got 100 files, and I need to find a bit of text in them and spit it out to a summary table. I might write a throwaway script that does that, 20 lines, it gets filed with the project and I never see it again.

5 NURBS is an abbreviation of Non-Uniform Rational B-Spline, a mathematical means for defining a surface or curve entity used in CAD modeling.
6 Munge is an acronym for 'Modify Until Not Guessed Easily'. It refers to a substitution process, commonly used for protecting passwords, in which letters are substituted for others (i.e. e = 3, or s = $).
However he also notes that this was not a common way of working:

It’s not a common way of working though, not even for people who do use Radiance™. It grew out of having to use Radiance™ differently. Most of the time people use it to generate 2D pictures of daylight factors, but I needed it specifically to predict radiation levels on a 3D mesh, and that lent itself to a different way of working. There’s one other guy in London who uses the scripts, but it’s certainly not common.

Mr. E described a different method, which was to add the additional information and properties required by structural analysis software to geometry within CAD environment:

We have written an interface with Sofistic™, which is the most powerful RC (reinforced concrete) design software. At the moment Sofistic™ relies on a clumsy and unstable AutoCAD™ interface which crashes all the time. Now we’ve managed to create any kind of surface and transfer it to Sofistic™ to analyse it. It’s going from crude Rhino™ to smooth Rhino™ with all the data attached. I’ve also developed interfaces to GSA™, so I don’t really use GSA™ anymore, instead I create what I call extended geometric models in Rhino™. It doesn’t need to be Rhino™, it could be anything. When you convert geometry to a structural model its all the same – you have element properties and you have supports and loads, it’s kind of generic data. The thing is that we don’t have to rely on any clumsy
interface within analysis software, we just have to deal with linking a powerful structural engine with a decent modelling tool, scriptable or whatever other way.

SYNOPSIS
This section has extended the theme of differing perceptions into the practical realm, by connecting it to the problems architects and engineers encounter when transferring design information from CAD to analysis software. As the disciplines have specialised and multiplied, so have their software and currently, transfer requires significant manual rework and ‘cleaning’. I have described some aspects of this problem as it is encountered and understood in practice. From the interview data, as well as design literature, I have advanced the argument that use, or the purposes to which digital representations are deployed, impacts upon the ability for design information to be transferred so that it is immediately useful. Because of this, I have chosen to investigate this issue using a pragmatic, in-situ (interfacing) approach, which raises the following claim:

A pragmatic approach to translation, whereby meaning is attached to operational consequences, can enable interdependent working between architects and engineers within the early design phase.

I will test this claim through practice-based project work within Chapter Six.
5.2 SHARED AND CREATIVE PROBLEM SOLVING: PERFORMANCE-BASED DESIGN

In The Social Life of Documents (Seely Brown & Duguid 1996), the authors contrast two images of documents: that of documents as darts with that of documents as a medium for negotiation and a means for reaching shared interpretations. As darts, documents carry preformed information or ideas through space and time, acting as a ‘conduit’ where information is put in at one end and taken out at the other. As a medium for negotiation, documents underwrite social interactions and coordinate practices (Seely Brown & Duguid 1996).

From my observations of practice, the sequential approach to architect engineer interaction treats 3D digital representations as ‘darts’. They have one originating author, the architect, whose CAD tools “give shape to a building and, in the process of documenting it, are the first to create original data” (Bazjanac & Kiviniemi 2007: 165). To enable an interdependent approach to architect-engineer interaction, which is my research aim, architects and engineers need 3D representations that “help architects, designers and engineers think critically yet qualitatively from different viewpoints” (Shea et al 2003: 553).

Within Chapter Four, I have connected the theme of shared and creative problem solving to processes of negotiation and evaluation. Drawing on the literature, I have stated that this requires close-coupled design processes and new tools and approaches that are suitable to the early design phase. The design results of these processes and tools neither belong to one discipline nor fully join different disciplines. This section examines an emerging problem solving methodology, performance-based design, in which architects and engineers are design co-authors. Within this area of practice performance, which can be interpreted differently across architectural and engineering fields, is viewed as a design driver that is on par with other processes of form making (Kolarevic 2005: 195). Performance-based design makes use of new 3D digital modeling and analytic tools and requires a close-coupling of the architectural and engineering design processes. It is of significance to my research because it enables a new means for converging traditionally separate aspects of design exploration by engaging architects and engineers in an iterative and reciprocal process of shared and creative problem solving.

Whilst performance is of shared interest to both parties, different disciplines focus upon different aspects of building performance. Within architecture, performance can be understood as a dynamic response, for instance glass that changes from clear to translucent (Whalley 2005: 30), or may relate to programmatic arrangement or site utilization. For quantity surveyors, performance can relate to minimising costs and economy. The different engineering disciplines may define performance as a function of structural efficiency, economy, environmental control, sustainability or constructability, to name just a few factors and work to deliver the most effective possible performance. All these aspects of building performance can be simulated and measured using 3D modeling and analytic tools, which consequently make available a new common ground of overlapping interests for architects and engineers.
5.2.1 ENABLING PERFORMANCE-BASED ANALYSIS AND DESIGN

The underlying idea of performance-based design is to “engage in a design practice based on feedback loops between making design decisions and evaluating their environmental impact, as a way to inform the on-going process of design” (Caldas & Norford 2002: 173). Advancements in computer-based 3D modeling and analysis methods underpin this possibility by enabling architects and engineers to simulate virtually many of the different building performances described above (Flager & Haymaker 2007: 625). However, up to very recently the use of computers to simulate building performance was limited to the detailed design stages (Kicinger et al 2005: 1943), whilst non-digitally based proscriptive or prescriptive analyses were used extensively throughout the earlier design phases of design.

Computational building performance simulations are termed descriptive (Augenbroe 2005: 99), as opposed to non-digitally based proscriptive or prescriptive approaches to analysis. Hartog et al (1998) and Mahdavi and Pal (1997) have addressed the definition of these terms in detail. The proscriptive approach to analysis sets out defined and encoded rules that limit the scope of acceptable design actions. Examples of this approach include building regulations, standards, codes and guidelines. Provided that designs stay within the rules, they are acceptable. The proscriptive approach has been criticised for potentially inhibiting design and innovation, as its simplicity leads it to contain “some rather limited concepts [that] should be regarded as anachronisms” (Lam & Mahdavi 1995: 442).

Prescriptive approaches to analysis, which “are utilised constantly in building engineering, as our processes for performance based design are in their infancy” (Schwitter 2005: 114), follow a predefined set of actions or standard rules of practice. Maver provides an example of the prescriptive approach from the context of service engineering: “after the design was complete, the engineer would, with appropriate ‘factors-of-safety’, multiply the area of the building envelope by a ‘U’ value, add the product of air-change-rate by building volume, double it, and choose a boiler twice that size” (Maver 1987: 48). As Schwitter observes, the “prescriptive approach may provide reduced risk, but it can also lead to reduced gains … in commercial practice, this approach is dominant as it is more time efficient and less encumbered by the unknown” (Schwitter 2005: 114).

The descriptive approach to analysis, which is made possible through the use of 3D digital modeling and analytic tools, enables a performance-based approach to design. Descriptive approaches to analysis require an explicit design description to be generated and a particular performance aspect simulated, and then for the engineer to interpret the analytic results, and rely on “an engineer’s training in creative problem solving and applying first principles for design” (Schwitter 2005: 114). As observed by Mitchell, “these procedures need not rely on unsubstantiated assumptions and rough approximations as in the past, and the simulations can model the performances of structural and environmental systems much more reliably” (Mitchell 1999: 840).
Performance-based design approaches shift design emphasis “from what a building is to what it does, defining the first by means of the second” (Leatherbarrow 2005: 7) by converging the traditionally separate explorations of the qualitative and the quantitative. As noted by Leatherbarrow, “there is no need to rank these two in a theory of architectural performance; important instead is grasping their reciprocity and their joint necessity” (Leatherbarrow 2005: 18). Within performance-based design approaches, design focus “shifts to the processes of form generation based on performative strategies of design” (Kolarevic 2005: 195), however these creative, close-coupled processes of negotiation and evaluation remain difficult to implement during early design studies. As Schwitter observes, “the challenge to today’s engineers involves seeking real ways of moving computational tools from simply being a means of proving design ideas to being integral parts of the design process, parts that can provide design input quickly and iteratively” (Schwitter 2005: 115). A key and historic aspect of processes of form generation based on performative strategies of design is the use of the tools generatively within the design process, in an active rather than reactive capacity.

5.2.2 EARLY COMPUTATIONAL PERFORMANCE-BASED DESIGN

Whilst the development of 3D analytic software is a relatively new phenomenon, computational research into generative performance-based design processes has a long 2D history. The earliest research into the use of the computer in the generation and synthesis of design solutions occurred in the area of spatial allocation (Eastman 1975: 9). This research explored how the computer might support the design process by proactively (generatively) investigating performance-based aspects of design that related to spatial allocation. Applications included the design of buildings such as schools, factories and hospitals. The techniques were applied at an early design stage, and typically addressed the problem of solving optimised planning layouts that minimized the distance and frequency of trips for a set of occupants (Frew 1980: 165).

In addressing allocation problems, it was assumed that “rooms that have a lot of traffic between them… should ideally be only a short distance apart” (Cross 1977: 33). An optimum solution therefore minimised the overall figure for journeys multiplied by distances. Examples of such programs include the Whitehead and Eldars™ program (Whitehead & Eldars 1965: 127) which designed single story building layouts by optimizing circulation patterns (Fig.56), and AIDATM which developed minimum cost house plans, maximized sunlight, view and privacy. Frew (1980) and Cross (1977) provide extensive overviews of this field.

Typically, the framework of space allocation programs consisted of two elements – the representation of the initial physical problem and the potential solutions, and the process by which the problem was transformed into a solution (Eastman 1975: 6). As I have noted previously in this section, this process was one of optimisation. After the program had proposed an optimised design solution, the human designer would then further adapt and develop it. In their 1975 example, Gero and Julian’s interactive building planning program begins its interaction with the designer with the somewhat antagonistic statement “Good morning… we are going to do a design
together. I will do the design first and then you can have a go at modifying the design to either improve it or to meet additional requirements that you, perhaps, didn’t consider initially” (Gero & Julian 1975: 201).

An example of the application of space allocation methods is the redevelopment of the Royal Canberra Hospital. The program used was the Topaz (Technique for Optimum Placement of Activities into Zones) planning model, which was developed at the CSIRO Division of Building Research (Crawford et al 1980). A relatively advanced program, Topaz incorporated a graphic CAD interface and analysed for multi-story buildings. The program solved a quadratic assignment spatial allocation problem to compute the optimum allocation of a set of activities over a set of zones within a series of time periods. Input data included activities, zones, time periods, expected future levels of activity, and an interaction matrix which described the amount of interaction between and within activities. A recognized difficulty with the model was being able to obtain an interaction matrix that was unbiased by the existing building layout (Crawford et al 1980: 60). The model output was either an evaluation of a layout (Fig.57), or the generation of an optimal layout and its cost.

These programs failed to gain a foothold in architectural practice. Their common characteristics were the difficulty of information input and output (Asanowicz 2002: 35), the narrowness of the design problems that they addressed and their inflexibility with regard to any criteria other than that used in the problem description (typically distance, weighted

---

7 This method is commonly referred to as ‘hill-climbing’.
‘affinities’ or trip times) (Weinzapel & Handel 1975: 61). Thus the results were overly determined and only optimal in terms of a very narrowly defined condition. Maver, in reviewing these early programs, makes the general note that “what is optimism from the point-of-view of travel efficiency might be pessimism from all other points of view” (Maver 1998: 8). With the machine designing, and the architect struggling with data input that was difficult to record, code and punch, as well as numeric outputs difficult to understand until converted into conventional architectural plans and elevations, there was, as reported by Cross, “a growing tendency amongst research workers to talk of computer-aided building design, rather than architectural design” (Cross 1977: 59, his emphasis). Not informed by the engineer’s historically documented interest in efficiencies and an economy of means, processes of optimisation led to the feeling that the results are “going to look like shit – guaranteed – and it will only produce environments that machines or machine-like people will want to inhabit” (Kahn, in Cross 1977: 59).

5.2.3 CURRENT BARRIERS TO SYNTHESIS

Increasingly sophisticated software and particularly the move from 2D to 3D representations have resolved many of the representational issues that made the early spatial allocation programs unattractive to designers, however others remain. As noted by Flager and Haymaker, “the potential for this technology to inform early-stage design decisions has not been fully realized because current tools and processes do not support the rapid generation and evaluation of design alternatives” (Flager & Haymaker 2007: 629).

Key problems, which I will now briefly describe, include translation (which has been examined in the previous section) and consequently iteration, assumptions about the use of the tools, and the actual and perceived appropriateness of performance-based design processes to the early design stage.

Analytic tools are primarily designed for evaluation rather than generation or synthesis. Malkawi (2005) and Chaszar et al (2006) have provided in-depth descriptions of the tools and processes used in simulating building performance, noting that these tools are typically single domain tools, developed around specific domain-related questions and algorithms which relate to thermal flows, structures, lighting, acoustics etc. Their use is limited by the difficulties of data transfer and their often challenging interfaces and inherent complexity (Chaszar et al 2006: 99).

Within the current model of use, “there is a range of digital analytic tools that can help designers assess certain performative aspects of their projects, but none of them provide dynamic generative capabilities yet” (Kolarevic 2005: 195, his emphasis). Many of the tools are ‘high resolution’, demanding significant levels of input, and “only experts can precisely understand the data and hence are always required to interpret them” (Malkawi 2005: 92). As Flemming and Mahdavi observe, “many programs require comprehensive input data (which is usually not available at the early design stage) and commonly do not assist the designer in terms of expert knowledge (input preparation, parametric analysis, result interpretation)” (Flemming & Mahdavi 1993: 162). Koutamanis notes that, far from assisting designers, the problems of integration may actually hinder them. Although “the combination of intuitive and quantitative evaluation offers a platform of effective and reliable communication with other engineers who contribute to the design of specific
“aspects” (Koutamanis 2000: n.p.), re-integrating the analytic results increases the already substantial quantity of information that the designer must deal with. Connecting to ideas developed within the previous section, Koutamanis continues that “the complexity of integrating analysis and synthesis suggests that the relationship between the two is not as direct as normative approaches have led us to believe. Integration requires interpretation and transformation capabilities…” (2000: n.p.). This means that design processes must develop feedback loops that are more than just the “juxtaposition of representations” (Koutamanis 2000: n.p.). Mahdavi and Pal also locate this problem as significant: “the performance program still needs to address the data inflation problem, as advanced (typically dynamic) computational tools for performance prediction commonly generate massive (and sometimes unmanageable) quantities of behavioural data” (Mahdavi & Pal 1997: 231).

These not insignificant factors have left the “full integration of simulation within the design process far from complete” (Malkawi 2005: 93), and within current practice the use of building simulation is “often regulated to the back end of the design process merely as a confirmation step to gain a quantifiable measure of the performance of the designed facility” (Lam & Mahdavi 1995: 439). When performance simulation occurs within the design process, it does so not in a synthetic and generative way but continues the assumption that design precedes simulation and evaluation, and that the tools, when used, are limited to evaluative purposes only: “the software, however, operates at the systematic level in the same passive fashion as two or three decades ago” (Kolarevic 2005: 198).

SYNOPSIS
I have introduced the concept of performance-based design as a means to explore architect engineer interdependency and extend the theme of shared and creative problem solving. Performance-based design is a method for shared and creative problem solving which makes use of new 3D digital modeling and analytic tools and a generative approach to architect engineer design exploration and synthesis. I have identified the underlying idea of performance based design, which is to engage in design practices based on feedback loops between making design decisions and evaluating their environmental impact, as a way to inform the ongoing process of design (Caldas & Norford 2002: 173). I have also identified several of the historic and contemporary limitations to performance-based design, which I will now connect to the experiences and understandings of practice.

5.2.4 THE EXPERIENCES AND UNDERSTANDINGS OF PRACTICE
The interview extracts within this section cast further practice-based detail on the preceding arguments and to supply clear evidence supporting the more theoretical observations outlined above. The typical interaction between processes of designing a building and simulating its physics are related in detail by Mr. B, who describes the time it takes to build a model for simulation and analysis, and the impact or otherwise that the results have on the architect’s design process:
It generally starts with a workshop session, sitting around a table with a pen and paper, running through some ideas, and the architect might come up with an idea like “ok, I want a glass roof”, and then we would respond, saying “that’s ridiculous, it’s an art gallery, you’ve got to control the amount of solar energy that gets into that space”. So then we would batter around ideas of how we might incorporate different types of shading and glazing systems, and from that generally generate an exciting idea, like this idea of the slats over the glass roof, and then everyone would say “wow, that sounds amazing, how can we actually make that work?”, at which point we would do a series of sketches that illustrate the idea, and then tell them that we will need to do some analysis on it, to test whether or not it is going to work.

At that point we would say give us a week or two weeks, go and build the model, run our iteration, develop a strategy as to how we will work out whether or not it’s going to be feasible, run the analysis, get the data together, analyse it and from that work out is it a goer or is it not? From that we would produce some sketches illustrating the solution of range of solutions, because they are quick and easy to do, and using the software we can do some screen-grabs. At that point we would put together a feasibility study, or otherwise we will send through the images in a short email. And that would go to the architect and they would respond either positively or just not take it into account at all. Say we had 6 options, they might say ok let’s just look at these two, and refine these further. At that point we would go back to the model, do deeper analysis or finer grain processing, and probably at that point do some serious checking to make that what we’ve modelled is scientifically ok, and then back and forth until we come to a solution that works. That process is always the case.

Mr. B

The value of establishing a quicker, more iterative and reciprocal relationship, and the reasons for the historical lack of such a relationship, is noted by Mr. C:

This iterative bit is where the value is. Historically with the type of work that I do, it was much smaller, possibly non existent. CFD has been really slow. You might have a kick off meeting with a client, and they say this is my room, model it for me, and the model wouldn’t be finished until 3 days before the report was due, because it was just so hard or the tool was so clunky. So you’ve got an answer, you report it and maybe some questions get raised, but you’ve actually finished by then.

Mr. B describes another factor that impedes interdependent interactions. The standard outputs of current analytic tools (visually descriptive colour plots and mathematically descriptive tables) remain difficult to integrate back into the design and CAD environments. This is a crucial limit to achieving a generative synthesis between CAD and analysis programs, as Mr. B observes:

The tools are useful to produce images, to illustrate: if we are talking about geometry in design they are very useful because you can print out what it looks like, and they can understand it, but when it comes to things like how a space is actually affected by a design they’re no good... [In this project] what we generated was graphs and hand sketches that illustrated how the space would perform in
Another significant issue is that fact that the design and analytic workflows are not well integrated. The self-described workflow of an analyst using Radiance™ (Radiance™ is an industry standard ray-tracing tool used in lighting analysis) is shown in Fig.58. This diagram is drawn by the author from information detailed by Mr. C. It can be seen that while playing a significant part in the process, digital tools and techniques other than PowerPoint™ play little role at the interface between architectural and engineering domains, and moreover the interface that exists is actually very limited. The transfer of information takes place primarily through verbal or graphic means, including false colour images (a technique for visualising datasets), tables and graphs, and sketches (the simulation tools are generally limited to the output of either calculations or false colour images). This information is typically packaged as 2D images and graphs within a report. The work of relating the design and analytic frameworks is carried out entirely by the engineer, and only a limited amount of intelligence is passed between the parties.

The lack of integration has led authors such as Kolarevic to argue that “the current simulation tools are completely useless from a design perspective. If you agree with that position, then you have to ask what will make them useful” (Kolarevic, in Kolarevic 2005: 234). Maver has also made this argument, issuing an early call for “software tools for the evaluation of the technical issues which are relevant at the conceptual stages, as opposed to the detailed stages, of design decision making” (Maver, 1998: 47). A potential solution is the development of ‘low resolution’ simulation tools, easy enough to be used by architects to provide a rough guide. As Kolarevic observes, “providing a certain degree of representational integration across a range of ‘low-resolution’ performance simulation tools is a necessary step for their more effective use in conceptual design” (Kolarevic 2004: 48). This view, however, does not seem to be universally shared by practice, and Mr. D argues that such tools may actually be counter-productive:

At the moment there is a tendency, which I am not so happy about, to simplify – to use simplistic software to calculate the energy of your proposed design. You have an extremely sophisticated spatial modeller, which allow for double curved nurbs surfaces to be defined architecturally, which includes an extremely simplistic solver for energy. The information that comes out of that will be wrong and therefore counter-productive. The other belief is that if you have the software you know how to adjust all the parameters, but that's not the case.

Mr. D

Kolarevic himself notes that “the lack of usable “low resolution” tools is further compounded by the expected degree of the user's domain knowledge and skills” (Kolarevic 2005: 198), a concern that, like that of Mr. D, centres on the non-specialist's ability to interpret correctly the information returned from analysis and to use it effectively in guiding design development.
The information flow between architect and daylight analyst. Self-described from an analyst within the Melbourne Arup office.

**Figure 58.**

The information flow between architect and daylight analyst. Self-described from an analyst within the Melbourne Arup office.
It might be argued that a factor silently underlying Mr. D’s viewpoint is that engineers are wary of risk taking and being found to be wrong, and therefore disparaging of low resolution 3D digital simulation tools. However, within the context of my investigations into interdependency, the area of low resolution tools represents a potential dead end. While such tools may seem to superficially support the intersection of architectural and engineering concerns by allowing architects to explore and become conversant in areas of design that have previously been the exclusive domain of engineers, they do so at the expense of the engineers themselves. Throughout this research I pursue an alternative approach to the problem of resolution that foregrounds interdependent working across the architect engineer boundary. The basis of this approach is the idea low resolution tools do not address the real issues of architect engineer interdependency, and that a better solution is to develop low resolution methods by which to use high resolution tools. Such an approach coincides with Malkawi’s observation that “rethinking the use of these tools from analysis to performance-based active design support has not been explored fully” (Malkawi 2005: 91). It also coincides with approaches present within practice, but represents a significant shift from typical modes of working and, as Achten has claimed, engages the engineer in processes that need “to be differently appraised by other design participants” (Achten 2002: 10). As Mr. G argues, there is no inherent problem with investigating a low resolution problem via high resolution tools, provided that there is an understanding of the evolving role that the tool will play in the design process:

Using precision tools is not a problem, provided you don’t expect to stay with those decisions – you just need to get started and then adjust from there. There are two sides to this: you want to be able to optimise very quickly and efficiently and know what you are working with, but you also want to free up the opportunities for investigating things that you would never normally dream of going near.

Mr. G

5.2.5 OPTIMISATION OR OPTIMALLY DIRECTED EXPLORATION?
It was noted above that the early spatial allocation programs pursued optimised solutions (Frew 1980: 165). In providing a clear definition of optimisation, along with many examples, Papalambros states that “broadly speaking optimisation means improving or fine-tuning the design in terms of one or more performance aspects. However, there is a very specific technical meaning of ‘optimisation’ as a rigorous mathematical statement. The basic assumption behind such a statement is that design process is viewed as a decision-making process, whereby one selects the proper functional form among many alternatives” (Papalambros 2002: 939). Processes of optimisation continue to drive many contemporary design generation systems, though Jannsen et al have drawn attention to the fact that “many of these systems tend to be overly focused on finding the generic ‘optimal solution’ rather than allowing for the development of ‘my proposal’” (Jannsen et al 2001: 137).

As Cross has noted, “so often there is no certain way of achieving an optimum solution to a design problem” (Cross 1977: 72, his emphasis), and even within tightly constrained optimisation processes,
“different runs of the optimisation tools may lead to different solutions with similar performance” (Caldas 2002: 174). This is particularly true within the early design phase, as Caldas observes: “at the early stages of design there is often no optimum solution, instead there is a large range of possible solutions all potentially having a good performance in responding to the problem under consideration” (Caldas 2002: 174).

More recent approaches to optimisation within performance-based design have been able to facilitate the exploration and navigation of a performance space (Luebkeman & Shea 2005: 17). Luebkeman and Shea describe such a space, which makes available a range of solutions, as “a set or ‘point cloud’ of optimized designs from which good designs can be selected based on preferences among performances, or viewpoints on the design” (Luebkeman & Shea 2005: 17). Each of these designs are thought of as optimally directed or, to use Watanabe’s term, ‘aptimized’ (Watanabe 2005: n.p.). Watanabe argues that optimal designs are only optimal in relation to a given set of constraints, and are therefore relative solutions which may or may not be absolutely optimal and are thus better thought of as “highly apt or appropriate” solutions. The term ‘aptimized’ combines the concepts of apt, and optimal, design solutions. From Watanabe’s architectural viewpoint, “it is possible to have any number of good things, but it seems difficult to call one of them “optimal” unless it is actually the best” (Watanabe 2005: n.p.); this view is sensible if one considers the number of factors that impact upon only the structural design of a building (Fig.59). My research will pursue optimally directed or ‘aptimized’ methods for design.

Figure 59. The range of loads that can inform the design of a structural system (Image source: Schueller 1986)
5.2.6 CONTEMPORARY APPROACHES TO PERFORMANCE-BASED DESIGN

Recent examples of design that has engaged aspects of building performance to drive generative feedback loops have had to involve rethinking of the use of simulation tools “from analysis only to analysis and synthesis” (Malkawi 2005: 87). They have also involved a rethinking of the design process so that it can engage in feedback loops, process integration and interpretive frameworks (Malkawi 2005: 87). While Kolarevic’s statement that “there is a range of digital analytic tools that can help designers assess certain performative aspects of their projects, but none of them provide dynamic generative capabilities yet” (Kolarevic 2005: 195, his emphasis) remains broadly true, a small number of designers and researchers have successfully explored this approach. A selection of this literature follows.

The potentials of Evolutionary Structural Optimisation8 (ESO™) for engaging architects and engineers in dialogue have been described by Burry et al (2004). They report the use of a structural optimisation technique, Evolutionary Structural Optimisation, to reverse engineer Gaudi’s design for the Sagrada Familia’s Passion Façade (Figs.60-61). The paper describes how the architect-engineer dialogue was achieved through a shared, iterative process of determining and refining the constraints and inputs for the ESO model. Interestingly, the authors find that this process was in fact reductive in nature, and that by reducing the initially large quantity of explicit starting parameters, the ESO process resulted in models that more closely matched Gaudi’s

---

8 The process of Evolutionary Structural Optimisation (ESO) involves gradually removing inefficient materials from a structure so that the residual structure evolves towards the optimum.
representations (2004: 316). As the authors note, the full potential of this technique as an integrative, generative tool is dependent on starting conditions that are not overly constrained, and this demands a better understanding of how flexible, early design intention models can intersect with generative, rule-driven processes (2004: 317). While unexplored in this paper, one can only assume that this is even more so for designs that are 'live'. Xie et al (2005) provides an overview of the ESO process and its extension, BESO (Bi-directional ESO).

Sasaki (2005) has used BESO in a ‘live’ design context in the design of the Florence New Station (Fig.62) with architect Arato Isozaki. The ESO method is extended by introducing two new elements, isolines (three dimensional isosurfaces) and bidirectional evolution. The resultant process is called Extended Evolutionary Structural Optimisation. By introducing bi-directional evolution the extended ESO process permits the restoration and growth of material, rather than only the removal of under-stressed material. Similarly to Burry et al, the authors describe architect-engineer interaction and integration occurring through the shared process of refining design constraints – giving “feedback by amending design variables until a satisfying shape was obtained” (Sasaki 2005: 81). As described by the authors, these design constraints are narrowly limited to a consistently flat roof, defined points of support and a minimum distance between the top and bottom chords (2005: 77), which reinforces Burry et al’s conclusion that a reduced set of explicit parameters may extend the potential for integration.

A different approach for linking analysis and synthesis begins with the definition of a structure and informs its configuration by a process of optimisation. Shea et al (2003) explore this approach when combining a structural optimisation tool (eifForm™) with parametric modeling software (Generative Components™) to support architect engineer negotiation using the example of designing a stadium roof truss system. The authors identify the problem that “the exchange of information between tools requires more than just transfer of geometric data” (2003: 555), an issue that I have examined within the previous section. They explore how a parametric early design model can use parameters and constraints as ‘production rules’ to describe design intent and direct the generative process. They describe a closed loop between the two programs, both of which are actively involved in the design process. The loop can be briefly summarized as follows. Planar truss elements are initially defined parametrically in Generative Components™, this information is passed into eifForm™ where optimized truss configurations are generated, which are returned as parametric geometry to the Generative Components™ model for evaluation. This loop iterates when the parametrically defined stadium changes configuration. One aspect that makes Shea et al’s research particular is that it recognises that negotiation is typically not all encompassing, but rather occurs around particular aspects of a design.

A hybrid approach is employed by Hemberg and O’Reilly (2004) to explore how structural analysis can be incorporated into existing generative early design tools. They raise the issue of computational cost that is associated with Finite Element structural analysis (FEA), particularly when analysis is performed iteratively, and forward the idea
that geometric criteria can substitute for structural analysis during early design. They propose a two stage strategy where, during an iterative, generative process, the performance of complete structural analyses (Fig.63) is separated by several iterations guided by a genetic algorithm (Fig.64). The authors describe a study in which structural analysis within ANSYS™ informs the fitness criterion that drives the form generation process, however it is unclear exactly how the tools interact or what the design intention is. It is suggested that such an approach demands that the user become responsible for understanding and interpreting structural performance, and “by inspecting the outcome, learn[s] how to adjust [the parameters] in order the desired result” (Hemberg & O’Reilly 2004: 4).

In exploring the relationship between design and lighting analysis, Caldas and Norford (2002) have used genetic algorithms (GAs) to optimize the size of windows in a building for lighting, heating and cooling performance during the intermediate to late stages of design. GAs are used to search for design solutions in a goal directed manner, generating possible designs, evaluating them and continuing the search guided by the simulation results. The authors contrast this approach to that of simulation, or “the designer generating a solution and subsequently having the computer evaluating it” (2002: 174), which they describe as “a slow and tedious process and typically only a few scenarios are evaluated from within a large range of possible choices” (2002:174).
Caldas and Norford study generative solutions for window configurations to the same building configuration in two different locations, one a heating and one a cooling-dominated climate. In contrast to the other examples, this method does not seem to have included design constraints – the authors note that in one of the solutions, windows were found to be too small to provide an adequate view. However, in generating many optimal or near optimal solutions, they suggest that “the use of an optimization tool like the one proposed here may not only provide increased energy savings but also introduce a positive degree of variability in the design” (2002: 183).

An increase in potential design options is also one outcome of a project reported by Luebkeman and Shea (2005), who describe the design of building envelope optimised for lighting and energy criteria. The project is for a media centre which houses several programs each with differing lighting requirements, and the optimisation process employed generates façade configurations (Fig.65) by testing for daylight factor against several test points, as well as for view and cost and thermal performance. The variable within the process is the type of façade panel, which can be of four types (opaque, clear, diffuse and shaded). The optimisation process configures these different façade panels into near optimal solutions for lighting and energy criteria (Fig.66) which balance the differing goals (Luebkeman & Shea 2005: 21) as well as functioning as a significant architectural ‘device’ which gives identity to the design.

SYNOPSIS

Within this section, I have examined the concept of performance-based design as a method for shared architect-engineer problem solving. This concept extends the theme of shared and creative problem solving which, in the previous chapter, was characterised by creative approaches to negotiation and evaluation, close coupled design processes and the development of new tools and approaches that are suitable to the early design phase.

I have defined performance-based design as the idea that the simulation of building performances should play a role in guiding design exploration via feedback loops. I then identified, through reference to literature and practice, current and historical limitations to this approach. Interview excerpts revealed that the time taken to construct a model for simulation, as well as the lack of integration between the analytic results and other aspects of the design process, were limits to design iteration. The practitioners whom I interviewed revealed that iteration and feedback, which form the basis for performance-based design, were valued qualities. Two possible methods of facilitating a more iterative and interdependent design process were examined – these were categorised as ‘low resolution tools’ and ‘low resolution methods for high resolution tools’. I have made the claim, to be tested through practice-based projects in Chapter Six, that:

‘high resolution’ analytic tools can support ‘low resolution’ methods for generative, performance-based design exploration
5.3 COMMUNICATION: BACK-PROPAGATING DESIGN INFORMATION

The role that representations play in facilitating the communication of design information in particular ways has been well described by Robbins within the anthropological study *Why Architects Draw* (Robbins 1994). Within this book, Robbins explores architecture's relationship with the drawing as simultaneously an instrument for internal dialogue and as a means for social activity and production. In exploring these issues, Robbins' central question concerns the drawing's primary role: should it be viewed as firstly a conceptual tool, as those he interviews would have it, or as firstly a social instrument, a role he argues is not sufficiently recognised by architects?

As a social instrument, Robbins describes how drawings facilitate particular types of interaction with others, both opening up and closing down dialogues between different parties: “the drawing allows the architect to compose a design, to orchestrate it, and to conduct the many players that will realize it” (Robbins 1994: 300). Robbins depicts drawing as a tool for linking what architects do with others, and as a key means for managing the process of design as it moves from conception to built reality. He also describes them as the key means for defining and reinforcing hierarchies that place the architect at the top: “as this process involves a socially hierarchical division of labour, the drawing plays a critical role in defining one's place in that process and the means through which that process is controlled” (1994: 298). The primary way by which drawing, the language of the architect, does this is through its role as the shared platform for discourse. Robbins states that “it is their [the architect's] medium and a form of language or discourse over which they have the greatest command and understanding… whatever the intent of the architect, setting up one's own discourse as a central instrument of communicative interaction sets limits, defines agendas, and creates social hierarchies” (1994: 297).

Robbins’ text reveals that the way information is represented can act to facilitate or preclude certain types of interaction and information flow. While he observes a hierarchy that places architects at the top and contractors at the bottom, within the context of this section we could substitute the word ‘start’ for top and ‘end’ for bottom. Robbins’ focus is on the designer as a disseminator of design information, and therefore making use of one type of representation to facilitate an outward flow of that information. This research investigates how conditions of interdependency can better inform the early design phase, and is therefore concerned with the inward and lateral flow of information. In the previous chapter, I identified that efficiency in communication facilitates interdependent working, and that communication needs to occur at multiple ‘levels’ to provide information supplementary to the minimum. It was suggested that this supplementary information might include insight into design goals of others, the reasoning behind them and the intentions of the designer. We can ask what representations might best facilitate this.

5.3.1 BACK-PROPAGATING DESIGN INFORMATION

In the past, as Mitchell has noted, “architects frequently found that they could sketch configurations that they could not describe sufficiently precisely or analyze sufficiently reliably, and therefore could not build” (Mitchell
The present day use of inherently precise 3D digital models to develop and describe designs has resolved many of the problems associated with describing complicated buildings; as Sharples and Sharples have discussed: "a difficulty with complex or non-predictable geometries has been that of envisioning and documenting the three-dimensional configuration in its entirety, with accurate and reproducible dimensions, a problem which is resolved with the use of computer modeling" (Sharples & Sharples 2006: 29).

Surprisingly, despite this fact, "some of these more complicated geometries are sometimes actually driving deeper wedges between architecture and engineering" (Schwitzer, in Kolarevic 2005: 234). This is because while aiding the representation and description of designs, 3D digital models do not necessarily make designs 'analysable' or 'buildable'. They do not necessarily make the process of interaction more flexible or more likely to avoid the "tremendous struggle between the structural engineer and the architect" described by Candela (Candela in Faber 1963: 14).

A crucial aspect of this struggle between architects and engineers is fabrication: "although architects today are not typically responsible for ‘means and methods’ of construction, those factors influence design, which must be buildable according to the logic of fabrication and erection processes" (Sharples and Sharples 2006: 29). Because design does not directly involve the late stage activities of making or doing, but precedes these activities, "it necessarily has to be predictive in order to anticipate what the consequence of the ‘making’ or ‘doing’ will be [and to develop this] with a well developed sense of premeditation" (Aish 2005: 10).

The sequential approach, which involves a designer who specifies the design, and a contractor who holds a detailed knowledge of building components and fabrication methods and who knows how to interpret the designer’s information in relation to those deployments, precludes premeditation or anticipation. To anticipate downstream requirements within an evolving design, detailed information needs to be drawn into and deployed within early design exploration. As Cornick and Mather have noted, the challenge is to communicate this information in an understandable and deployable format – "which might not necessarily be drawings" (Cornick & Mather 1999: 118).

This idea that drawings might not be the best means to communicate detailed downstream information into the early design process (Cornick & Mather 1999: 118) is significant to my research aim of enabling interdependency in architect engineer interaction. Obviously, information that better supports the ability of architectural and engineering designers to integrate fabrication and other downstream constraints such as performance simulation into design exploration may not be easily representable within digital 3D models either. While Szalapaj has noted that "the secret, if there is such a thing, with digital tools in practice, lies in the way they support connections between the design of complex sculptural forms and the rational methods of fabrication and construction that are needed to realize them" (Szalapaj 2005: 10), Maher and Burry have equally argued that simply substituting 3D models for drawings does not automatically guarantee a coherence of information (Maher & Burry 2006: 202), let alone an efficiency of communication or representation.

Within the context of the early design phase, Fekete has observed that "the main objections on the designer’s part towards such practice [using 3D models during the early design phase] concerns the difficulties associated with modeling the loosely defined information so typical of the early design stage" (Fekete 2003: 116).
Synthesising this loosely defined information with highly defined downstream requirements has the potential to further impose upon the designer. Causing additional difficulties is the idea that “the sharing of information, particularly in a process that, at its best, involves collective conceptualisation, is complicated by the very close and reciprocal relationship between the partial knowledge about the object of design and the mode of expression or representation of these ideas” (Burry et al 2005: 288). As Sharples and Sharples express this same idea, the logic of making needs to be “embodied both in the construction process itself and also in the means with which the design is communicated” (Sharples & Sharples 2006: 29).

5.3.2 THE EXPERIENCES AND UNDERSTANDINGS OF PRACTICE
The interview extracts within this section cast further practice-based detail on the preceding arguments and to supply clear evidence supporting the more theoretical observations outlined above. Identifying the importance of fabrication and buildability to the architect engineer interaction, Mr. E describes one role of the engineer as that of bringing knowledge of how things are built to the design process:

Often what we do is to introduce a level of rigour. An architect might say that they want to do Voronoi, but you look at it and it's not actually Voronoi, or they might want to do a weave or smooth stuff but it's never really that. And the issues of construction are never addressed: how do you take it to the fabricator, how do you describe it, how do you realise it? In those cases we introduce a notion of geometric rigour and knowledge of how things are built. Sometimes it is quite hard – the architect might be very hooked on the image that they've sold to the client. It's one thing to win a competition, and another to make a project. Both are interesting, but it's not just a matter of making that image.

The idea that the design has already occurred, and may have already been ‘sold to the client’ is a significant part of the problem, according to Mr. A. Amongst those interviewed, Mr. A focussed most extensively on issues of construction and fabrication. Having been involved in several projects in the U.K. and America which have challenged traditional design-bid-build approaches, he viewed this area as extremely important for architect engineer interaction. Succinctly describing the current condition, Mr. A stated that:

Typically, a design is produced and tendered without the input from the people who are going to manufacture and erect it.

As projects are being delivered far more quickly, the expectation is to deliver them quickly – there's never time to explore these things once the contractors on board, its bang, bang, bang, deliver, deliver. So the expertise of these people, which is quite considerable, is often lost – it's a fait accompli by the time they are on board.

My thoughts since being here is that Australia is a few years behind the UK in quite a number of
fields, things like the Egan report are now being successfully implemented in the UK, but it means throwing the procurement process up in the air and rethinking the whole thing, and trying to bring the expertise of all parties on board, and informing the design process.

Mr. A described how, lacking design input from fabricators and manufacturers, the design process and ultimately the building suffers from too little knowledge of cost and of smarter and more efficient design strategies that may exist:

Another example is two stadia in the US, both myself and the architect put an awful lot of work into the geometry of the stadium bowl, which was pre-cast concrete, and again, tender process, tender return, and the [fabricator] came back and said “it’s a real shame that you didn’t make those 4 things exactly the same by tweaking the radius by a few degrees, and instead of 14 different things you could have had 4”’. Great advice, but it came too late in the day, and far too late to change the design because the program was so tight.

Things come in too expensive and people are forced to re-evaluate or value engineer. But then the pressure really is on because you’ve spent your fee, so you haven’t got the resources to commit too much to the process, and the scope for innovation is taken away quite considerably.

A significant factor that Mr. A identifies as being problematic is that, not knowing who the fabricator will be, there is no ability to tailor design information to gain possible efficiencies. He argues that this problem is linked to tendering processes:

There is definitely a hole there, in the process, and we’ve talked at length in many jobs about “ok, what’s the format in which we’ll produce our data?” It really depends on how the contract is going to be let – if it gets let to tenderer A, you know the way their process works so you can tune your design to be in accordance with that.

But one fabricator will have his machinery set up to do one, and another to do other. And you can bet your bottom dollar that if you design one way the guy who wins the contract will have his machinery set up to do the other. But if you knew in advance what that was going to be, you could tune your design specifically for that.

Finally, Mr. A locates significant benefits, including cost and time savings, to bringing information about fabrication and other downstream aspects of design into the early design process:

There are benefits for engineers and certainly for architects to really understand how things go together, particularly complex forms, and how the initial design assumptions can be made to work far more efficiently.

Most architects and most of the engineers I work with talk the same language, generally, they talk a
different language to manufacturers and a different language to contractors, typically. That part of the dialogue normally works reasonably well - it's the rest of it – how do you move from an idea to a representation in whatever format to something that's real. That's where the cost savings can be, that's where the reduction in procurement times can come to the fore. It is in understanding how the design becomes a reality, and how you can model the cost, the constructability of it, where I see the difference being made over the next 2 or 3 years. A couple of the projects I have worked on have made a big impact on that. Definitely the model is there.

5.3.3 CONSTRAINTS, REQUIREMENTS AND 3D DIGITAL DESIGN TOOLS

It is not within the scope of my research to investigate alternative procurement models to the depth that such an enquiry would require, although The Travellers project, reported in Chapter Six, will address this in practice. However, the idea that drawings might not be the best means to communicate detailed downstream information into the early design process (Cornick & Mather 1999: 118) is significant to my research aim of enabling interdependency in the early design explorations of architects and engineers. Alternate representations need to facilitate the involvement of often missing expertise (see the above interview extracts from Mr. A) to provide information supplementary to the minimum. In the previous chapter, I suggested that this supplementary information might include insight into design goals of others, the reasoning behind them and the intentions of the designer.

What is needed, in addition to 3D models, is a way to efficiently convey and embed design requirements as intentions and constraints that guide the development of geometry. As noted by Shelden, “much more is known internally about the nature of products than can be exposed to the design process simply through the occupation of space. The notion is that design or engineering intent generates occupancy of space in a given building configuration and that this intent can be coordinated in a much more direct manner” (Shelden 2006). Shelden locates the capacity to convey design intent in software tools that can express intentions independent of geometry, meaning that they persist over geometric variation: ‘it is this capability that allows the conventional notions of the linearity of the design process to be reversed, that late stage decisions can be potentially back propagated upstream into design iteration’ (Shelden 2006). Expressing or declaring design intentions within these software tools makes them explicit, a quality that Simon has linked to communication across boundaries:

“The ability to communicate across fields – the common ground – comes from the fact that all who use computers in complex ways are using computers to design, or to participate in the process of design. Consequently, we as designers, or as designers of design processes, have had to be explicit, as never before, about what is involved in creating a design and what takes place while the creation is going on”

(Simon 1969: 137)
Fekete has observed that “the main objections on the designer’s part towards such practice [using 3D models during the early design phase] concerns the difficulties associated with modeling the loosely defined information so typical of the early design stages” (Fekete 2003: 246). As Williams has described, designers generally use rules and constraints in the development of a design, but these are rarely made explicit:

“When a person designs an object they will consciously or unconsciously adopt a set of rules. These may be some rules of proportion or the principles of structure or fluid mechanics, or a limitation on cost or the materials available. The rules are extremely unlikely to be in the form of an algorithm; they will be vague, incomplete, contradictory, open to dispute and require a great deal of intelligence to interpret. One of the main functions of the professions is to make sure their rules are so complicated that only their members and their expensive software can interpret them” (Williams 2004: 79).

Currently available tools that allow working in this way are parametric modeling software and scripting. Parametric and associative modeling tools developed in the late 1980’s, and while previously restricted to the aerospace, automotive and shipbuilding industries, and at the periphery of design practice, parametric modeling is now a feature of many CAD packages9. Rather than define geometry explicitly in relation to a single coordinate system, parametric modeling allows flexible geometric definition through the use of constraints, relations and parameters. Geometric entities can be constrained dimensionally or associatively to other entities or input data, and controlled by manipulating that geometry or changing input variables. Parametric models are thought of as flexible models because they are free to move within the constraints of their parametization (Burry et al 2001: 77) - as input variables change, geometric change is propagated through the system via dependencies.

Within parametric software, relationships, parameters and constraints are generally defined graphically. When scripting, the designer defines these same entities as a text-based set of rules or instructions and is, in a sense, interacting with the internal workings of the computer (Burry 1997: 492) albeit on a very low level. Most CAD software has an endemic scripting language, and within these programs scripts can be used as tools for automation and/or tools for generation. As Loukissas and Sass explain, “scripts can make use of functions already coded into the parent software environment or add new functionality. There are limitations to what can be scripted in any given software environment. However, scripting provides just the right amount of access to underlying structures, which allows one to represent personal and project specific design processes” (Loukissas & Sass 2004: 177). A significant difference between parametric models and those generated by scripting is that the scripted models are not dynamic.

As an automator, “the script’s role [is] to eliminate digital draftsmanship by collapsing the process of geometric modeling” (Dritsas 2005: 706). Used generatively, “concepts are expressed as generative rules so that their...

---

9 Current examples include Revit™, Bentley Sturctural™ and MicroStation™ through Generative Components™ (GC), Rhino™ through the Explicit History plugin.
evolution may be accelerated and tested. The rules are described in a genetic language which produces a code-script of instructions for form generation” (Frazer 1995: 9). In either application, scripting can be used to embed design intentions and constraints into the process of design exploration: “the written script is an explicit representation of this intention” (Loukissas & Sass 2004: 176).

Using Visual Basic for Applications™ (VBA), scripting can also play an important role in addressing another limitation of CAD software which prevents downstream information being used within the early design process. Although CAD tools have allowed the precise description of geometric information constructed within any particular software environment, used ‘out of the box’ they have not permitted the user any easy interpretation of design information coming into that environment. In part, this is because of their limited ability to import digital information, a function of the limited models of users and the problems of use held by the developers of CAD systems (Henderson 1998: 139). CAD software typically affords a number of ways of saving a file, both through the various native file-types and exchange formats, through an ‘EXPORT AS’ type command (Fig. 67). Generally the user chooses whichever file-type is native to the program that they will be exporting to, if available, or else chooses a certain file-type because it provides some particular affordance. For instance, .OBJ file type might be chosen because it provides a high degree of control over the representation of curved surfaces.

CAD programs do not have the explicit inverse of this command, ‘IMPORT AS’ – all importing is done through the all eggs in one basket ‘IMPORT’ command, where whatever
information within the file that is semantically compatible with the program is extracted in a single way, and that which is not, ignored. Contrast this to opening a text file (.txt) with Excel™ (Fig.68), an exercise in constructing a mapping that may provide some insight into how an ‘IMPORT AS’ command in CAD might function. Via the import wizard, the user can determine exactly how the information within the text file should be mapped to the Excel™ spreadsheet: on what line the translation should begin, whether the relationship of text strings to cells should be determined by fixed widths or delimitation, how those delimiters should function, whether the cells should be formatted to reflect dates or times etc, etc. There are a surprisingly large number of possible string to cell mappings.

SYNOPSIS
Within this section, I have raised the idea that back-propagating design information is an important element to enabling architect engineer interdependency. This concept extends the theme of communication, which in the previous chapter was linked to the ideas that efficiency in communication facilitates interdependent working, and that communication needs to occur at multiple 'levels' to provide information supplementary to the minimum.

Drawing on the available literature, I have proposed that the traditional mode of design representation, 2D drawings, is not an effective way of integrating detailed information into the early design phase. Interview excerpts revealed that information related to fabrication and making is typically not available within this phase of design, and that as a result the work of both architects and engineers suffers, as does the project outcome. They also revealed that this information is not generic, but is tied to specific ways of working and to specific fabricators. Key benefits of integrating detailed fabrication information with early design exploration included the efficiencies of data transfer, and more significantly the early informing of initial design assumptions that effect both architects and engineers.

I have introduced to two alternative and related methods: parametric modeling and scripting. These tools provide a means to technically support the suggestion, made in the previous chapter, that the 'supplementary' information often required includes insight into design goals and intentions of others. Both approaches make it possible to explicitly represent intentions and constraints, which I have identified as an efficient means of crossing boundaries. This investigation, drawing on literature and interview extracts, leads to the claim that:

Parametric design and scripting can facilitate a back-propagation of downstream information into early architectural and engineering design exploration

This claim will be tested through practice-based projects in Chapter Six.
5.4 TRUST: REPRESENTATION & EXPLICATION

In many cases, the authors referenced in Chapter Four noted that trust was a crucial aspect of successful interdependent working. Those from the field of organisation theory noted that a lack of trust limited the ability to achieve optimum outcomes in interactions with others (e.g. Tyler & Kramer 1996), and those from the field of architectural design located mutual respect and trust as fundamental to collaboration (e.g. Ritchie 2001). From this literature, I have deduced that trust is a crucial site of intersection when working across the architect-engineer boundary.

Within Chapter Four, I introduced two ideas. Firstly, that trust is a means for managing risk and uncertainty, and secondly that communication plays a crucial role in developing and sustaining trust in temporary groups. As Hardy et al observe, “in an inter-organisational relationship, trust grows out of a communication process in which shared meanings develop to provide the necessary foundation for non-opportunistic behaviour. Accordingly, trust can be conceptualised as a communicative, sense making process that bridges disparate groups” (Hardy et al 1998: 69). This section further develops the theme of trust by exploring the idea that 3D digital tools can facilitate trust by supporting alternative approaches to representation.

Working interdependently demands that “almost invariably we are working with others whose skills we lack, or have only to a limited degree” (Thornton 2007: 102). This interaction occurs at a time when, as Eastman observes, “the easy, close-working relationship between designers and builders has largely disappeared” (2004: 20). Close and overlapping relationships between designers and builders have been replaced by a design-bid-build method of contracting that many argue promotes litigation and limits innovation (Eastman 2004: 21).

From a computational design perspective Brazier and Wijngaards have described the role of trust in distributed, collaborative design: “human participants in a distributed design setting often know whom they trust, and whose abilities they value. This knowledge is not often made explicit. It does, however, influence distributed design processes (i.e. the way in which members of a design team assess and incorporate each others’ designs, objectives, evaluations)” (Brazier & Wijngaards 2002: 71). However, from organisation theory we know that “organisations are moving away from formal hierarchical structures to more flexible and temporary groupings around particular projects” (Tyler & Kramer 1996: 8), and that the traditional, rational model of trust is not completely applicable.

As Rousseau et al observe, “trust is not a behaviour (e.g., cooperation), or a choice (e.g., taking a risk), but an underlying psychological condition that can cause or result from such actions” (Rousseau et al 1998: 395). This description of trust, along with those previous, make it understandable why trust is a highly difficult factor to address yet simultaneously crucial to interdependency. In focussing on architect-engineer interaction, I will define trust as a means for facilitating interdependency by managing and reducing apparent risk. The benefits of trust are well described by Swan et al, who state that “the reason to trust for most organisations is that it can lead to faster, cheaper projects. Costs of problem solving are reduced, and litigation, that can lead to total breakdowns in relationships, can be avoided” (Swan et
al 2002: 4). While the issue of trust recognises that technology alone is not enough to support interdependent working (e.g. Kvan 1999, Ritchie 2004), it does not imply that technology cannot play a role in supporting this outcome.

5.4.1 TRUST IN THE ARCHITECT ENGINEER RELATIONSHIP

The generation of trust, or ‘buy in’, by all parties is critically important to the success of any project, particularly those that are challenging. When trust is lacking, design processes are less open and the reliance on checking mechanisms and conventional, formalized procedures increases. Bennett & Peace have described the general lack of trust that currently characterizes the construction industry:

“The construction industry had a structure based upon the perceived status of the various professions and trades. But it provided no explicit coordination or control. Consultants fiercely maintained their independence, contractors competed for work and specialists struggled to maintain the integrity of their skills against market driven demands for lower costs and faster delivery. Clients dealt with an industry that appeared chaotic by using competitive tenders and tough contracts to protect their own interests” (Bennett & Peace 2006: 7)

whilst Hesselgren provides a snapshot of the detail as it affects the work of architects and engineers: “BDP seems to have handed over their geometry generation to their structural engineers, a huge mistake by their architects I think, geometry is design” (Hesselgren, 2005).

As Baier has stated, when we “allow many other people to get into positions where they can, if they choose, injure what we care about, since those are the same positions that they must be in order to help us take care of what we care about” (Baier 1986: 236). All of the three sites of intersection that I have established and explored so far within this chapter (differing perceptions, shared and creative problem solving, and communication) can only occur when architects, engineers and others share discipline specific knowledge with one another to enable better outcomes. However, as Cornick and Mather have pointed out, “knowledge sharing will only come about if each team member feels that in doing so they will directly benefit. Observation of project practice through much applied research indicates that with the ‘adversarial’, traditional method of procurement there is a tendency for team members to use their knowledge to defend their position as this is what they are so often forced to do” (Cornick & Mather 1999: 119).

An important reason for team members seeking to defend their own position is that they have been placed in a situation of uncertainty and risk: “the contractor uses his or her construction ‘knowledge’ once site production has begun to create all sorts of claims to overcome the in-built design inefficiencies” (Cornick & Mather 1999: 121). Uncertainty arises because of design inefficiencies, as Cornick and Mather note, but also because of uncertainties that stem from inefficiencies in design representation. As

10 Building Design Partnership (BDP) is an U.K-based architectural practice
Smyth states, “trust is needed over things you cannot see, things not known” (Smyth 2006: 97). Inefficiencies in representations can occur because of inaccuracy: “although drawings, specifications and schedules from the contract documents are often assumed to be complete and free from error, the reality is that they are not” (Emmitt & Gorse 2007: 8). They can also arise though a lack of understanding, which is independent of the accuracy of the representation. To guard against this uncertainty, representations are often reworked: “it is rare that fabricators actually use the model that we give them. They might use it as a basis for their explorations, for developing their own models that they can trust” (Schwitter, in Kolarevic 2005: 234).

5.4.2 THE EXPERIENCES AND UNDERSTANDINGS OF PRACTICE

The interview extracts within this section provide access the real life experiences and opinions of practitioners within Arup, and are presented here to extend the more theoretically derived ideas presented above through reference to practice-based understandings. The benefit that increased openness and shared meanings between the different interacting parties could bring to the process and the end design result was discussed by Mr. B:

“As designers, we are all working to achieve the same endpoint, and if everyone knows a bit more about how they do it, and what they do, the increased awareness of everyone’s issues is beneficial to the end result: either in terms of you get there quicker, or you get there with better quality.”

Providing evidence for Hesselgren’s statement that control of the geometry equates to control of the design (Hesselgren 2005), Mr. I describes how a lack of control over the geometry can filter down into a lack of trust in accepting design information, in this case the engineer’s 3D models. In doing so, he adds another reason for information duplication to those which I have stated previously in section 5.1 (i.e. single domain tools, specialist representations and ‘cleaning’):

“It’s all dictated by the site constraints, and is geometry driven. We dictated the geometry and the form, and they had a say in what they wanted to do with the cladding. They lost interest in that, because they weren’t driving it, and it became a case of them trying to get us to do their work for them. They weren’t geared towards working in this way – we would give them a 3D model and they would produce their elevations and plans. They would redraw these rather than take them from the model.”

However, illustrating the potential communicative role that 3D digital tools might play in addressing this complex issue, Mr. A describes how such tools are an important means for understanding and managing apparent complexity:

“I think 3D certainly helps, the greatest aid is in understanding the complexity; for the kind of things that we do, understanding complexity is half the battle. There are a number of structures that I’ve worked on in which it’s very difficult to represent those in a 2D form, and even a simple 3D representation you get a much better appreciation of how the thing might go together, how you
might rationalise it to make it easier to build, easier to form. So there are certainly benefits in that respect.

5.4.3 TRUST AND COMMUNICATION
Many designers may “consider that an adversarial attitude serves them or their organization better” (Thomas & Thomas 2005: 68), and much current discussion focuses on developing new contractual forms of trust (e.g. Thomas & Thomas 2005, Construction Task Force 1998). Of particular interest to my research, however, is the way in which the problem of trust (managing risk and uncertainty) can be specifically addressed through representation. Shelden (2002: 25), in discussing some of the reasons why Gehry Partners has pursued non-conventional methods to document and communicate design, explicitly addresses the relationship between representation and the barriers to ‘buy-in’:

“While fabricators could build the shapes, the process of bidding and coordinating the projects presented difficulty to construction managers. Accuracy of quantity takeoffs could not be guaranteed using conventional methods of measuring off the plans. Shop drawings – necessary for describing the detailed fabrication geometry – were difficult to render into orthogonal views… The limitations of understanding the project geometry through the lens of two dimensional views exacerbated perceptions of project complexity”
(Shelden 2002: 25)

This view coincides with that of other practitioners. Sharples and Sharples of Shop Architects11 note that “complex geometries cannot be adequately described using conventional drawing methods, which rely on orthographic projection of three dimensional objects. Plans and sections are a kind of shorthand notation representing typical conditions” (Sharples & Sharples 2006: 29). Swan et al state that “if team members can produce information that is clear and accurate, and the other members can rely on it, then uncertainty will be reduced” (Swan et al 2002: 12). Rice has described working in this way on Arup projects – “it is part of our procedure, in designing unusual buildings, to explain precisely what it is we’re doing… so that [all parties] understand that they aren’t taking exceptional risks” (Rice 1991: 104). He states that “professional liability, and the problems that it appears to create, are effectively a myth, and that the real problems lie elsewhere” (1991: 104). Together, these statements suggest that representations can facilitate trust, and therefore mitigate risk, by increasing understanding.

The more complicated a building is, the less typical conditions will meaningfully describe it, and the less any kind of accurate understanding can be guaranteed from them. This limits any kind of shared meaning. When the representations used to communicate design information do not facilitate an easy or accurate understanding of the design, other parties cannot be certain of the level of risk that they are entering into. As Smyth notes, “where project uncertainty is assessed as a probability from the data, yielding an initial perceived risk, the conditions of trust can potentially reduce this

11 SHoP Architects are a New York-based architectural practice.
risk by building confidence’ (Smyth 2006: 106). Shelden’s statement, with those of Rice and Mr. A, highlight the role that explicatory processes and representational techniques which make it easier to understand and utilize design information are certainly central to managing and reducing apparent risk.

SYNOPSIS

Within this section, I have raised the idea that representations can facilitate the building and management of trust by enabling explicatory processes and representational techniques which make it easier to understand and utilize design information. This concept extends the theme of trust, which I have described within the previous chapter as an important mechanism for managing risk, uncertainty and reducing apparent complexity within teams characterised by the temporary alliances of autonomous partners.

While recognising that, because of trust, technology alone is not enough to support interdependency, in this section I have established that 3D digital tools and representations might enable approaches that facilitate trust and therefore interdependency. The literature and interviews raised several key aspects relating to representation and explication, which included increasing the awareness of how and what others are doing, the idea that representations can exacerbate or decrease apparent complexity, and that they might increase understanding and ‘buy in’. This investigation, drawing on literature and interview extracts, leads to the claim that:

3D digital design tools can enable representations that facilitate trust by increasing understanding and reducing perceived risk and uncertainty
SUMMARY
This chapter has extended the four sites of intersection identified within Chapter Four through reference to literature and extracts drawn from the interview of nine Arup practitioners. This material has identified how each of the four sites represents crucial aspects in enabling architect engineer interdependency. Each site has been located in terms of problems particular to the use of 3D digital design tools and the potential of such tools for enabling architect engineer interdependency. Each site has been given context and meaning through a relationship to practice-based experiences and understanding.

Site one: Differing perceptions
As the disciplines have multiplied and specialised, so has their software. Architects and engineers use different digital tools and representations to facilitate their exploration of a common design problem. Currently, transferring design information between these tools (CAD and analytic software) requires significant manual rework and ‘cleaning’, which limits their initial use and subsequent iterations. Enabling more effective translation would therefore enable interdependent architect engineer design exploration. I have advanced the argument that the purposes to which this information is deployed, together with insight into the engineer’s work process, impacts upon the effectiveness of translation. I have chosen to investigate this site of intersection using a pragmatic, in-situ (interfacing) approach, and raised the following claim:

* A pragmatic approach to translation, whereby meaning is attached to operational consequences, can enable interdependent working between architects and engineers within the early design phase.

Site two: Shared and creative problem solving
Without differences between the parties, the range of possible exchanges would be nonexistent. Shared and creative problem solving occurs through negotiation and evaluation, which requires close coupled design processes and new tools and approaches that are suitable to the early design phase. The design results neither belong to one discipline nor completely join different disciplines. I have raised the concept of performance-based design as a means to explore architect engineer interdependency through shared and creative problem solving.

Performance-based design is a method for shared and creative problem solving which makes use of new 3D digital modeling and analytic tools and a generative approach to architect engineer design exploration and synthesis. It allows architects and engineers to engage in design practices based on feedback loops between making design decisions and evaluating their impact, guiding the ongoing process of design. I have also identified several of the historic and contemporary limitations to performance-based design, which I have connected to the experiences and understandings of practice.
Literature and interview excerpts revealed that the time taken to construct a model for simulation, as well as the lack of integration between the analytic results and other aspects of the design process, were limits to design iteration. I have made the claim that:

‗high resolution‘ analytic tools can support ‗low resolution‘ methods for generative, performance-based design exploration

Site three: Communication
Communication is arguably the one aspect that pervades all the other sites of intersection (i.e. in the idea that efficiency in communication facilitates interdependency). I have examined one particular aspect of communication, the idea that back-propagating design information is an important element to enabling architect engineer interdependency. I have suggested that in interdependent working, communication which facilitates this activity needs to include insight into design goals and reasoning. I have also suggested that the traditional mode of design representation, 2D drawing, is not a highly effective mechanism for providing this insight and consequently integrating downstream information into the early design phase.

I have found that requirements related to fabrication and making is typically not available within this phase of design, and that as a result the work of both architects and engineers suffers, as does the project outcome. Key benefits of integrating detailed fabrication information with early design exploration included the efficiencies of data transfer, and more significantly the early informing of initial design assumptions that effect both architects and engineers. I have introduced to two alternative and related methods: parametric modeling and scripting. Both tools make it possible to explicitly represent intentions and constraints, which I have identified as an efficient means of crossing boundaries. I have claimed that:

Parametric design and scripting can facilitate a back-propagation of downstream information, relating to fabrication and potentially other aspects of design, into early architectural and engineering design exploration

Site four: Trust
The issue of trust recognises that technology alone is not enough to support interdependency. Trust facilitates interdependency by managing risk, uncertainty and reducing apparent complexity within teams characterised by the temporary alliances of autonomous partners. I have suggested that 3D digital tools and representations can facilitate the building and management of trust by enabling explicatory processes and representational techniques which make it easier to understand and utilize design information. The literature and interview extracts illustrated that awareness and understanding can exacerbate or decrease apparent complexity, and that those representations which enable more than the ‘typical condition‘ to be described might increase understanding and
‘buy in’. I have made the claim that:

*3D digital design tools can enable representations that facilitate trust, and therefore interdependency, by increasing understanding and reducing perceived complexity, risk, and uncertainty*

To further test the validity of this framework as a way of understanding how 3D digital tools can enable interdependent working between architects and engineers within the early design phase, I will continue my investigation by working on practice-based projects within the context of practice. These are reported in the following chapter, Chapter Six.
Chapter Six: Project-based Testing

6.1 DESIGN\textsubscript{(ARCH)} | DESIGN\textsubscript{(ENG)}
6.2 DESIGN | ANALYSIS
6.3 DESIGN | MAKING
INTRODUCTION

This chapter details eight practice-based projects that I have undertaken over a three year period at the Arup Melbourne office. My role within the projects has been that of primary and often sole 3D modeller and sole software scripter. In many of the projects, I have worked as part of a team involving two or more structural or mechanical engineers, often also interacting with designers in the architectural offices with whom Arup Melbourne is undertaking these projects. In projects that do not involve outside architectural practices I have assumed the role of architect.

The following three sections are labelled:

- Design_{Arch} | Design_{Eng}
- Design | Analysis, and
- Design | Making

The bar character that separates the two terms in each relationship denotes a concurrency operator, and indicates that the two processes are executed in parallel. Each section details one major project, followed by several minor projects in support.

The section Design_{Arch} | Design_{Eng} reports projects that use a common language of geometry to link architectural and engineering design ideas through geometric interpretation. The section Design | Analysis reports projects in which analytical tools have been used actively and generatively to guide and synthesise design exploration. These projects include both optimisation and form finding processes. The final section, Design | Making, reports a project in which streamlining the design process around fabrication constraints led to a changed relationship between engineer, architect and steel fabricator.

In detailing the following projects, I have been constrained to a certain extent by the fact that the source material is not fully available for further discussion as it has been generated in the context of live and commercially sensitive projects. Additionally, some of the architect-engineer communication can only be summarised as they also belong to the original party. However, the illustrations presented alongside each project represent primary source material, and are by-products of the actual work in progress not secondary documents or drawn after the fact.
INTRODUCTION

As I have noted, one of the main functions of early design exploration is to inform and guide the development design work without having to undertake the substantial tasks of detailing and documentation. Critical to this exploration are the issues of speed, iteration, change and flexibility. One limit to the architectural and engineering designer's capacity in each of these areas is that, as Holgate has observed, “the input offered [by engineers] is unnecessarily accurate, and too slow, to be useful at the early stages of the process” (Holgate 1991: 148). Another limit is that, as I have described, the construction of a 3D model for analysis takes time and precision; as the design changes these models usually require significant amounts of rework. As noted by Flager and Haymaker, the potential of digital tools to inform the early stages of the design process has not been fully realised because current tools and processes do not support the rapid generation and evaluation of alternatives. This means that few, if any, options can be adequately studied (Flager & Haymaker 2007: 625).

Synchronisation of architectural and engineering inputs is made difficult by the relatively low resolution nature of the early design phase, where both parties work with incomplete and imprecise data. The use of CAD within this phase is, as Fekete has observed, “judged simply not worthwhile... in the context of architectural design today, particularly during its early stages, the benefits of CAD are far from being that significant” (Fekete 2003: 246). He continues that “the main objections on the designer’s part towards such practice concerns the difficulties associated with modelling the loosely defined information so typical of the early design stages” (Fekete 2003: 246). But what role might CAD play to address these factors and facilitate interdependent design exploration? Can geometry address problems of differing perceptions, and establish a synthesis not just between data but also between design ideas?

At the beginning of the early design phase, engineers aim to “understand which are the main drivers within the decision making process” (Mr. A) and to then “get into the process by suggesting what works and what wouldn’t work” (Mr. C). This is a low resolution means of back propagating engineering intelligence into an emerging design; however the impact of this approach is limited by its form of representation. For some engineers sketches, often used in conjunction with heuristics or ‘rules of thumb’, are the main support – Mr. A mentioned that “typically hand sketches are the fastest and easiest way to communicate”, and that “getting one of the CAD guys to draft up what looks like a structural diagram isn’t a particularly effective way of communicating a thought”. According to Mr. G, “you have to build up those interactions, to find a way in to that process, because it is not normally led by the engineers anyway. It’s got to be something that the architects can work with”.

Between the extremes of the sketch and precise analysis, geometry is a language common to both architects and engineers. This section examines the use of generative and flexible modelling techniques to synthesise architectural and structural design parameters around the common (and low resolution) language of geometry. In generative models, the 3D model is a by-product of
the design process, which involves identifying and then encoding as a script a series of rules that define the 3D geometry. Flexible models are models in which geometry is controlled (but not necessarily generated) through rules and constraints. Both forms of modeling are able to support fast and highly iterative forms of design exploration. Over the three projects detailed in this section, flexible and generative models are used to capture architectural and engineering design intents and link these to geometry to support iterative and flexible design exploration. The first project, *Bendigo Canopy*, details an example of establishing architect-engineer interdependency that continues across different CAD software, whilst the second project, *Marina Bay Bridge*, examines the benefits of an already established interdependent condition. Lastly, the *School Canopy* project initialises an exploration into the interaction between architectural and engineering design software.
6.1.1 BENDIGO CANOPY
PROJECT DESCRIPTION
The design (2006) is for a canopy, one part of a larger outdoor mall redevelopment in Bendigo, Victoria. The canopy, 50 x 27m, provides a covered public space on the main street between several major commercial buildings (Figs.69&70). The project team was led by Rush/Wright Associates, a landscape architecture practice based in Melbourne. Arup’s involvement was limited to the structural design of the canopy. When Rush/Wright Associates approached Arup with the project they had already developed a 3D model of the canopy, which they had used in presentations to stakeholders, however they were open to development and interested in pursuing the idea of integrated form and structure.

The facetted form of the canopy is difficult to realise and its design was respondent to a complicated mix of factors. The research firstly focused on the development of a synthesising logic by which architectural and engineering design intents could converge geometrically. Subsequently, I developed a scripted tool which automated the application of that logic through the generation of 3D geometry within Rhino™ to support early formal investigation. The outcome of this script was a canopy design. This part of the research covers the period until the end of conceptual design phase, at which point the project ceased. The research then explores how the project might have logically progressed beyond this phase by integrating detailed structural analysis with flexible modelling, and concludes with the use of an optimisation algorithm within the parametric software CATIA™ to explore a shared design space.

RECONCILING TWO DIAGRAMS
Rush/Wright Associates’ design idea for the canopy drew significantly on Bendigo’s local context and history. They proposed a facetted, geologically inspired roof-form supported by a forest of tilted columns, ideas that emerged from Bendigo’s rural situation, gold rush history and promenade precedents (Fig.71). This design was provided to Arup Melbourne as a Rhino™ 3D model (Fig.72). Initial questions posed by Rush/Wright Associates concerned the sectional depth required by the roof structure, and the impact of this on the formal undulation and folding. Initial structural analysis of the design within GSA™, performed with generic assumptions and with the intention of understanding how the structure performed ‘as is’, showed that forces were greatest at the points where the columns met the canopy roof (Figs. 73-76). The understanding taken from this analysis was that these points required greater strength and structural depth.
These different understandings of the design were synthesised through the idea that there should be a relationship between columns and facets, and that the facets might function as capitols or tributaries, channelling force down into each column (Fig. 77). In structural terms, beam depth and the performance of the structure in terms of stress and deflection was related to the condition of support, leading to a highly effective structural system. The key criteria in developing the form were recognised as being:

- number and distribution of the columns
- proximity of the columns to the edge and to each other
- depth of structure required to span between columns or support the cantilever edge
- inclination of the columns

This synthesis provided the foundation from which a shared geometric syntax could develop.

DEFINING A ‘RULE OF THUMB’ BASED MODEL

Defining the facets through their relationship to the columns posed a problem: Rush/Wright Associates did not know what the column configuration would be, nor would they for some time. The mall area below the canopy was subject to complex pedestrian movements and requirements for access by emergency and maintenance vehicles. These and other similar factors impacted upon the column location. In addition, the columns needed to interact with street furniture and other design elements proposed by Rush/Wright Associates (Fig. 69). Many different configurations would need to be tested architecturally as well as structurally to determine the best design, and having to computationally test each configuration within GSA™ would have greatly limited the number of design iterations possible. One way of addressing this problem would have been for an engineer to sit beside the designer as they explored these non structural concerns, giving ‘gut feeling’ guidance on each iteration. While effective, the call-out rates for an average engineer make this method uneconomic. Instead, a set of rules that drew on structural rules of thumb were defined as follows to govern the extent, depth and ‘pointiness’ of each facet.

FACET EXTENT

The extent of each facet can be defined through a Voronoi diagram. A Voronoi diagram computes, for a given set of input sites, the set of all points closer to each input site than to any other, producing an inherently efficient tessellation of space. This set of points is called a Voronoi cell, a collection of such cells a tessellation and the boundaries between each cell in a tessellation ridges. In the context of the canopy, the Voronoi diagram computes an efficient shape for each facet. To determine this, the top point of each column is treated as an input site, a Voronoi diagram is generated1 and the resultant Voronoi cells mark the extents of each facet (Fig. 77).

---

1 This script adapted David Rutten’s Voronoi script, available at http://www.reconstructivism.net/
Figure 69. Architect's perspective of the proposed canopy (Image source: RWA)

Figure 70. Architect's plan showing proposed mall development (Image source: RWA)

Figure 71. Sample of imagery driving the architectural design concept (Image source: RWA)

Figure 72. The initial 3D model provided by RWA to Arup

Figure 73. Structural analysis: Deflection - Z

Figure 74. Structural analysis: Shear - ZX

Figure 75. Structural analysis: Bending moment - X

Figure 76. Structural analysis: Bending moment - Y
FACET DEPTH
The depth of each facet is a function of the distance from the top of the column to the furthest extent of the Voronoi cell. There are two possible conditions, depending on whether the cell lies on an edge of the canopy or not. In cases where the cell is not on an edge the depth of the facet is determined by the equation \( \text{span}/20 \). Where the cell is on an edge, if the equation \( \text{span}/8 \) for any point on the Voronoi cell is greater than \( \text{span}/20 \) for the furthest point, \( \text{span}/8 \) is used to determine the depth of the facet.

Using these rules the larger or more eccentrically shaped the Voronoi cell the lower the branching point of the column and the deeper the facet. Conversely, columns supporting only small areas of roof have higher branching points and shallower facets. The depth vector is defined by the vector of the column associated with that particular facet (Fig. 78).

LOWER SURFACE
The shape of the lower plate is also defined by the Voronoi cell, and its scale is a function of the distance from the top of the column to the furthest extent of the Voronoi cell. This surface is rotated so that it is normal to the vector defined by the column associated with that particular facet. The reason for having this surface is not structural but rather relates to buildability, in that it is a means to avoid all members converging at a single point.
DEVELOPING A 90% RIGHT STRUCTURAL TOOL

The application of these rules to a column configuration is enough to generate the information necessary to geometrically model the canopy (Fig. 79). It is interesting to note that, in rule-based systems such as this one, geometric complexity has no relationship to the amount of information necessary to define it — in fact the rules outlined above were determined within an hour through sketching on a single piece of paper. Apparent geometric complexity is driven by the column configurations, which are easy to control and reconfigure.

But while the introduction of a geometric logic synthesised architectural and structural design ideas and provided all the necessary geometric information, manually building the 3D model by following the rules remained a time-intensive task with a high scope for error. In particular, the generation of the Voronoi diagram is not something one could achieve quickly by hand. To reduce the time taken to generate each 3D model I developed a tool that, with minimal user input, automated the application of the rules and the generation of the 3D geometry. The tool took the form of a script, written in RhinoScript™, which was run as a button within Rhino™. It had several significant benefits. Firstly, the time to construct one model with an average number of columns is approximately 30 seconds, meaning that many different possible designs can be explored quickly (Fig. 84). Secondly, the geometric definition of the model was such that it could be imported directly into GSA™ without the need for any cleaning. Thirdly, using the script allows the designer to focus their exploration on non-structural parameters,
Figure 84. Geometrical outcomes from running the tool on varying column configurations.
in this case pedestrian movement, landscaping and street furniture, while generating results that are structurally informed and will not require significant structural adjustment at a later date.

The results of the use of this script (Figs. 80 & 81) were presented to Rush/Wright Associates, who subsequently presented the work to local stakeholders. The intention was to provide the ‘90% right structural tool’ to Rush/Wright Associates so that they could generate design variations, however this did not happen due to the project ceasing for reasons unrelated to Arup Melbourne’s involvement. The final design iteration produced was modelled by Arup using the tool (Figs. 82 & 83). The script is reproduced in Appendix A4.1.

CHANGING RESOLUTION
The project ceased as a live project within the office, but continued as a research problem which explored its implicit trajectory. This research asked ‘Had the project progressed, and computational analysis replaced rules of thumb, could it have continued to do so as a flexible model which connected architectural and engineering design intents?’ To progress in such a way demanded a continuing synthetic definition of form and structure, the use of an optimisation approach and for the possibilities of further design change, and quick investigation, to remain open. This required a shift in the software used, and occasioned a conceptual shift in the structural diagram.

The need to shift resolution from a rule of thumb model to a higher level of analysis required shifting programs, from Rhino™ to CATIA™. CATIA™ is a parametric modelling program which allows the designer to explore and control formal manipulation through declared parameters and constraints. It combines a number of different workbenches, each specific to certain tasks or disciplines, and allows the designer to work across those workbenches, sharing information between them. This ability to share information, particularly between CATIA’s parametric modelling, structural analysis and optimisation workbenches was a significant factor in choosing this tool to extend the research. An important precursor for this part of the investigation was the Parametric Bridge (Maher & Burry 2003 & Maher 2006), a collaborative project by SIAL, Arup Research and Development and Arup London Group 4. This project demonstrated how the Product Engineering Optimizer, a Knowledgeware function in CATIA™, could be used to drive formal change in response to structural analysis. Additionally, part of this Parametric Bridge project involved comparing the results of CATIA’s Generative Structural Analysis with Arup’s in-house structural analysis software GSA™, the result being a close correlation.

As described previously, a conceptual shift in the structural diagram occurred at this point. In the rule of thumb model, each column and facet supports only the small element of roof structure immediately above it, with which it forms an integrated cell. The intention was that these cells could each be fabricated in the shop and then be tied together on site. This structural diagram was replaced in favour of a single large trussed roof structure supported by all the columns. The conceptual difference is found in the supporting elements acting either individually or as an integrated system.
Figure 85. Parametrically varying the column branching point within CATIA
PARAMETRIC MODEL
The CATIA™ model comprises of a collection of line-based branching columns supporting a planar rectangular surface representing the trussed roof. The columns are variants of a single generic column, which incorporates a single ‘free’ parameter, the branching point. This is the point at which the column connects with the lower plate, from which it ‘branches’ to support the canopy. Each of the 36 columns has a separate parameter governing the position of this point, and each of these points is free to move along the vector of the column. There are 6 variants on the generic column, relating to the range in the number of sides of the Voronoi cells – from 3 to 8. Each variant was saved as a PowerCopy, and the PowerCopies instantiated multiple times to populate the model. Because the geometry and the free parameters within the model were so narrowly defined, much of the setup geometry could be imported into CATIA™ as explicit geometry. This geometry included the column centrelines and the Voronoi cells, which define the facet extents and the points at which the branches connect to the trussed roof. This geometry was imported directly from the ‘rule of thumb’ Rhino™ model as an IGES file.

STRUCTURAL MODEL
The parametric model was transferred to CATIA’s Generative Structural Analysis workbench where, working with an Arup engineer, a finite element (FE) model was created from it. This model was not validated in either of the two analytical software packages commonly used within Arup, being Strand7™ and GSA™, however recent research by Maher & Burry 2006 has included such a validation and suggests that a close correlation is possible. In FE analysis geometric entities are divided into small mesh elements defined by geometric nodes that can be described mechanically by analytical equations (Maher, 2006: 42). A FE model in CATIA™ is a set of representations that are complementary to the geometric model, and built on top of it. These representations aim to capture a sufficient amount of model specification to initiate a solution process, and consist of two parts. There is firstly a set of system definitions, which includes a Mesh objects set, a Properties object set and a Materials object set. These definitions relate to the density at which the geometry is meshed, the nature of each member (beam, flexible or rigid connection etc) and the physical properties attached to each member. Secondly, there is a set of environmental definitions, which include an Analysis Case, defining the type of analysis, the points of restraint and the load case, and a Solution set, which defines the type of results sought and their representation through images, reports and graphs.

OPTIMISATION
There were two possible approaches to optimisation². The first was to treat the geometry as given and optimise the sizing of members to reduce tonnage, an approach that has been applied on other projects within the Arup office for projects such as the Beijing Water Cube. The second approach was to treat the member sizing as given and to optimise the geometry to maximise structural efficiency and minimise tonnage. This second approach was chosen because

² While it is possible to optimise for both geometric shape and member size within the same optimisation method, the approach taken in this project focuses solely on geometry. The reason why one and not both areas were investigated stemmed from my direct experiences of a previous project, the Travellers. In that project, Arup undertook a design investigation that developed structurally optimal section sizes for the Travellers sculptures. The wider design team, including architect, project manager and client, found the solutions generated were unoptimal in proportion and aesthetic - a key issue being that the slenderess of the members diminished the legibility of the sculptures within the project's exposed urban setting (this finding clearly demonstrates that a solution can only be described as ‘optimal’ within the context of a particular issue, and the same solution may be unoptimal when judged against other criteria). Reflection upon this experience, and in particular on the similarity in urban setting of the Travellers and Bendigo Canopy, led to the decision to use a fixed section size that met proportional and aesthetic criteria, and to then optimise for geometric shape.
it represented an extension of the previous exploration, replacing design guidance through encoded ‘rules of thumb’ with design guidance through computational analysis.

Optimisation can be performed in CATIA™ through the Product Engineering Optimizer workbench, and can use 2 types of algorithms: local algorithms (Conjugate Gradient) and global algorithms (Simulated Annealing). For this project, simulated annealing was used. Within CATIA, this takes the form of a move and test cycle, whereby a design is analysed, then changed, then analysed again. The designer can define a minimum or maximum target value, and with each iteration the parameters designated as free are moved until the solution closest to the target value is found. Initially, optimisations for stress and tonnage were performed individually to observe the model’s performance.

STRESS
In optimising for stress, the designer is seeking to minimise the amount of stress in the structural system and to maximise the structural efficiency. Stress was measured in all the columns and branches at 200mm intervals. The optimisation routine aimed to find a configuration that reduced the highest recorded stress. The points at which branching could occur was limited to a range between 20% and 80% of the column height, and were initially configured to occur at 60%.

The result of this optimisation generally lowered the branching point where columns occurred at canopy edges and where they supported large spans, and raised the branching point where columns supported small spans (Figs.90 &
Figure 89. Histograms showing progression of the structural optimisation process and sequence of configurations tested
The progress of the optimisation routine is described in Fig. 89. Measured via CATIA’s™ ‘Principal Stress Tensor’ sensor\(^3\), the maximum recorded level of stress in the model was 198 MPa (Fig. 87), in contrast to 337 MPa in the non-optimised model (Fig. 88).

### TONNAGE

In optimising for tonnage, the designer is trying to minimise the quantity of material within a structure, and therefore the cost of that structure. In the CATIA™ structural model columns were defined as having the properties of a 273.1 X 12.7 CHS, with a weight of 64.6 kg/m, while branches had the properties of a 139.7 X 6.0 CHS and a weight of 17.9 kg/m. While the columns weigh more than the branches there are far fewer of them, and at a given point their weight plus that of the branches will be minimised. Similar to the optimisation for stress, the points at which branching could occur was limited to a range between 20% and 80% of the column height, and were initially configured to occur at 60%.

The results of this optimisation\(^4\) were that all columns lengthened and all branches shrank, so that the branching point for all columns occurred at 80% (Figs. 92 & 93). This indicated that an optimum result was not found, meaning that the most efficient configuration lay outside the acceptable 20% to 80% range. The recorded tonnage was 3.9 tonnes, in contrast to 4.36 tonnes in the un-optimised model.

---

\(^3\) The terms ‘Principal Stress Tensor’ and ‘Stress Principal Tensor’ are used interchangeably within CATIA™ and the CATIA™ help documentation; however both expressions are potentially misleading. Both variations of the term describe a measure of stress, which is a tensor with principal values and principal directions. As the term ‘Principal Stress Tensor’ is used within CATIA™ this term is also used within this thesis.

\(^4\) While multiple optimisation runs were conducted (an average of ten for each of the three investigations), the solutions described are selected from one run of each optimisation.
MULTI-CRITERIA OPTIMISATION FOR STRESS AND TONNAGE

Exploring optimum configurations for stress and tonnage individually led to an interesting observation. Optimising for stress generally brought the branching points for most columns down while optimising for tonnage moved them up. Including both parameters within one objective function made it possible to perform a multi-criteria optimisation. In multi-criteria optimisation the goals are typically conflicting, meaning that in the conventional sense one optimal solution does not exist, and the designer instead attempts to find the best trade-off among the conflicting objectives.

Initially, the objective function was defined as: \[ \frac{\text{stress}[\text{iteration } n]}{\text{stress}[\text{initial}]} + \frac{\text{tonnage}[\text{iteration } n]}{\text{tonnage}[\text{initial}]} \]. Minimizing this function, provided that the initial configuration was not already an optimum, finds a configuration that trades off the two objectives equally. This led to a solution with a maximum stress of 246 MPa, measured via CATIA’s ‘Principal Stress Tensor’ sensor, and weighing 4.15 tonnes (Figs 94 - 96).

FURTHER EXPLORATION OF THE INTERSECTION

At this point it was possible to produce optimally directed results for stress and tonnage, however a means for implementing further top down design intentions was lacking. A relatively simple mechanism was introduced, which was to allow the explicit definition of parameters, representing design intents, by using fixed values for certain branching points (Fig. 97). This designation allowed these columns to play an active part in the analytical and optimisation process but not free to change their branching
Fixed branching points along the short sides of the canopy within the CATIA model. After setting these design parameters, the optimisation process finds optimally directed solutions given this additional condition. A similar technique is used in other optimisation processes, for example ESO (Xie et al 2005), where fixed geometry is termed ‘non-design’ geometry and is part of the optimisation but cannot be removed. In the context of the Bendigo Canopy project, the design rationale for controlling the model in this way was the desire for thin edges, which provide a cleaner line and allow people to see further into the structure. For this to occur the cells along the front and back edges of the canopy, as experienced while walking along the mall, are set to a minimum value from which they are unable to change, and the optimisation process run again with all remaining parameters flexible. The results of this exploration are shown in Figs 98 & 99.

**PROJECT SUMMARY**

This project has demonstrated several means by which 3D digital tools can be used to intersect architectural and engineering design explorations throughout the early design phase. In its low resolution phase, the use of scripting to generate 3D models increased the ability to explore potential design solutions quickly. The input and encoding of structural engineering knowledge into the script meant that despite the flexibility of the approach, each potential solution had an inherent structural intelligence. In the high resolution phase of the project, the use of parametric modelling linked to structural analysis enabled the further flexible exploration of the design through informed geometric alteration.

From the architect’s perspective, the script enhanced the ability to explore design solutions,
providing a means to quickly visualise and evaluate designs. The structural intelligence inherent in each outcome would better position the architect to explore design issues other than structure, for example access requirements or the placement of street furniture, and to then evaluate the formal response. This evaluation would be made easier with the knowledge that significant later changes to the structure, and therefore the form, are less likely to occur. For the architect, the script functioned as a knowledge transfer mechanism, providing an insight into the thought processes of the engineer.

For the engineer, the use of a generative modelling technique enabled a more effective communication of structural design. Using 3D geometry, rather than false colour plots, calculations or 2D drawings, provided an immediate way for the architect to make sense of engineering intentions, parameters and logic through a language both parties understood. As for the architect, the script allowed the engineer a rapid means of generating the design and of ensuring that the geometry was such that it exactly matched later requirements for computational structural analysis within GSA™.

While the use of parametric modelling linked to structural analysis was conducted as speculative, rather than ‘live’ research, several observations can be drawn. This approach enabled the further flexible, integrated exploration of the canopy design to occur within a single 3D model, rather than as separate architectural and structural models. This eliminated the possibility for miscoordination between architectural and engineering designs. Within CATIA™, it was possible to investigate the possible trade-offs between the two engineering parameters, stress and tonnage, and the architectural preference for thin edges. The automation of this process using the Product Engineering Optimizer was necessary because of the complexity of the interrelation of these factors; attempting to determine optimally oriented results would have been impossible via 2D means.

The use of optimisation to guide the exploration was not intended to produce an entirely optimal structure, but rather one that was optimally oriented with respect to design factors and intentions particular to the project. The prior use of the scripted tool was of high benefit here; during the process of its development the major engineering and architectural parameters were made explicit. Subsequent construction of the parametric model, which in the early design phase can be a difficult process due to the undeclared nature of many design parameters, benefited greatly from knowledge accrued during the scripting process. This suggests that a hybrid scripting then parametric/optimisation strategy is of benefit in early design situations.
6.1.2 MARINA BAY BRIDGE
PROJECT DESCRIPTION

The Marina Bay Bridge project (2005), by COX architects, is a pedestrian bridge located in Singapore. The design of the bridge features counter-rotating spirals that are tied together and connected to the bridge deck, which they support, via ties and struts (Figs.100-102). Because of its geometry, the bridge has been nicknamed the ‘Double Helix bridge’. At the time of my involvement in this project, the bridge’s structural system had already been designed by Arup, but the centreline of the bridge was yet to be determined by COX architects. A bridge centreline provides the start point, end point and trajectory of the bridge. It is one of the base pieces of information from which a structural design is developed.

In this project, as in many bridges, form and structure are completely integrated. The design had been developed as a set of geometric rules which determined the bridge form and structure, and this approach had already engaged architects and engineers in a high level of interdependence. The investigation presented here is therefore not concerned with establishing interdependency but rather with exploring the practical benefits of this condition.

Within Arup, two issues had emerged as important. The first of these, emerging from concerns about analysis and documentation, was to find a way to make a 3D model of the bridge’s structural system that could be applied to the eventual true centreline without the need for any remodelling. The second issue, emerging from concerns about fabrication and standardisation, was to determine whether the members making up the bridge deck would
be repetitive or uniquely sized. To investigate these issues a 3D model was constructed within CATIA™.

THE GEOMETRIC RULES
Much of the rule set determining the bridge structure had already been encoded as a Visual Basic for Applications™ (VBA) script by Arup, which generated geometry in Microstation™. The significant geometrical variables for the bridge, which needed to be replicated in the CATIA™ model, were defined as follows:

- A variable radius of the centreline circle
- A variable number of bays
- A variable angle between bays
- A variable radius of main spiral
- A variable radius of minor spiral
- A variable distance of the main spiral centre point above the deck
- A variable deck width
- A repeating pattern of connections between the main and minor spirals

THE PARAMETRIC MODEL
The power of CATIA™ comes largely from the rigour of its geometric definitions, a property that makes for a relatively slower modeling process compared to many other CAD programs. The need to model quickly the 3937 elements eventually contained in the final model posed a challenge that necessitated a hybrid manual/scripted approach.

The manually modelled part of the model addressed only the creation of construction geometry⁵. This included a circle, whose radius was controlled via a parameter, from which a segment whose length was determined by the distance given by the number of bays multiplied by the angle between bays was extracted. The resultant arc, which represented the unknown centreline, was then tilted on an axis. Points, the number of which were defined by the variable number of bays, were then located evenly along the arc.

At each bay point, a plane normal to the centreline (defining a local Z axis) and a plane vertical in the global Z axis was created. The planes vertical in the local Z axis represented construction geometry onto which the hoops and spirals were oriented (Fig. 103), while the planes vertical in the global Z axis represented construction geometry onto which the bridge deck was oriented. It was the combination of these two coordinate systems, and their impact upon the repetitiveness or otherwise of bridge deck edge members, that catalysed this investigation.

---

⁵ Construction geometry is geometry which is created to enable the process of developing other geometric elements.
A PowerCopy, which is a reusable feature that contains geometry, parameters and relations, was then created. This contained a larger circle (C1), whose radius (R1) was controlled with a parameter, and a smaller circle (C2) whose radius (R2) was defined via the equation \( R2 = R1 \times \cos(30 \times (\pi/180)) \). The centre of the larger circle was offset upwards in the local Z axis by the distance \( R1 \times \sin(30 \times (\pi/180)) \). Using these equations, the points at which the two circles intersected were at the same global Z height as the bridge deck. The part of the circle C2 which occurred above the intersection with circle C1 was then divided into 12 segments, and the circle C1 divided into 16 segments (Fig. 104). This PowerCopy was then instantiated at each of the division points along the bridge, the circles oriented to the local Z axis planes (Fig. 105).

The geometry created until this point all represented construction geometry. The remainder of the model, consisting of spiral members, ties and struts and bridge deck elements, was scripted using CATIA’s Visual Basic script editor. A segment of this script is reproduced in Appendix A4.3

While apparently complicated, this geometry is highly repetitive in nature. The major spirals were generated by a script that made a line between point1 on bay1 to point2 on bay2 and so on. The minor spirals were generated in a similar manner (Figs. 106 - 108). The tie and prop members, which introduce a high level of visual complexity to the bridge, are of only four types (Fig. 109). These repeat along the bridge in a regular A, B, C, D sequence. Only 3 of these types are individual, as B and D mirror one another. A VBA script looped through the bays, firstly inserting tie elements and then
prop elements. Lastly, similar looping scripts inserted the geometry defining the deck edge, and cross bracing.

To test the model, CATIA’s replace function was used, which allows the designer to replace one geometric entity with another (Figs 110 & 111). This can be a very powerful tool in cases such as this, where all the geometry in a model is dependant on one entity (in this case the initial arc).

THE OUTCOME
Using the CATIA™ 3D model, it was possible to ‘apply’ the bridge to any centreline. This removed the need for any remodelling when the ‘stand in’ centreline was replaced by the true one. In exploring the bridge deck member lengths, it was found they were in fact different. (Fig. 112) This was tested by exporting the model to Rhino™, where a RhinoScript was written to measure each deck edge element and create a histogram (Fig. 113). The results revealed why the question had been posed – the difference in length between the longest and shortest deck edge members was below 2mm, and when building tolerances are taken into account this amounts to each member being exactly the same length.

PROJECT SUMMARY
This project demonstrated a significant benefit to using a geometric rule system to intersect architectural and engineering design exploration. The interdependency generated through this approach allowed engineering investigation to examine highly detailed problems at a point when basic design information was still lacking. Typically, engineers would not investigate
Figure 109. The four repeating tie and prop sections
Figure 110. Testing the script on a new centerline 1

Figure 111. Testing the script on a new centerline 2

Figure 112. Member lengths measured in CATIA

Figure 113. Histogram showing difference in length of bridge deck edge members

Figure 114. 3D print of bridge section 1

Figure 115. 3D print of bridge section 2

Figure 116. 3D print of bridge section 3
problems of this resolution until the design was fixed, because subsequent changes in design would invalidate that work. As a result, an investigation such as that conducted for this project would not be available to inform the development of the design. In a similar vein, the use of parametric modeling made the construction of a 3D model a feasible exercise – to manually remodel the complicated bridge geometry each time the design changed would have required significant amounts of time. The 3D model later formed the basis for a series of rapid prototypes which aided in the communication and understanding of the design (Figs. 114 - 116).
6.1.3 SCHOOL CANOPY
PROJECT DESCRIPTION
During the first three months of my Arup Melbourne internship I undertook a small, self-generated study project for the design of a school courtyard canopy (Fig. 117). The purpose of this project was to understand better the interaction between CAD and Arup’s in-house structural analysis software GSA™ via firsthand experience, and to gain insight into the tools and processes which were used by the engineers and analysts. An additional benefit of this study was the development of my relationships with engineers within the office. While the school canopy project explored interaction, and was defined so as to further the research question, it did so only in a very basic way. The CAD analysis interaction was deliberately limited to one-way information transfer and did not attempt to ‘close the loop’.

As an architect entering an engineering firm, I had used CAD tools in study and in practice but had never used analytical tools and had no experience with the information these programs required and the information they output. With this background, any research conducted through live projects would be severely limited by my lack of basic operational knowledge outside my own discipline (architecture).

GSA™ is used to simulate structural performance in the majority of projects undertaken by the Arup Melbourne Office. It imports geometric files from CAD software via DXF format, reads and writes a native GWB format, and also uses a GWA format, which is

---

6 When I entered the office GSA™ (General Structural Analysis) was used almost exclusively by the structural engineers within the Melbourne office to simulate structural performance, while another simulation tool, STRAND7™, was used at the Sydney office. Over the 3 years of study, STRAND7™, has become increasingly used in the Melbourne office.
To determine the new perimeter, duplicate edges are identified. Possible impact upon adjacent triangles is determined through reference to circum-spheres. Beginning the process of adding new panels - the user selects a new input point.

Possible impact upon adjacent triangles is determined through reference to circum-spheres. Final result.
openable in MS Excel™. The GWA file format, which records node positions, members, loads and restraints and the results of analysis, is often used within the design workflow for two reasons: A) to access MS Excel™ methods and B) because it is often easier to manipulate geometric information within a spreadsheet than in GSA™.

This project explored the problem of designing a panelling system for a courtyard pavilion. The hypothesis was that a panel design based on a Delaunay triangulation (Fig. 118) would result in geometry that was closer to an initial freeform shape while using fewer members and panels than a system based on a standardised triangulation (Fig. 119). The aim was for the designer to freely define this triangulation within an iterative process, guided visually by the shape and by feedback from structural analysis. This posed the question of how might structural analysis be used to guide the designer’s next move?

Using a freeform shape as input, I developed a RhinoScript that generated a ‘basic’ Delaunay triangulation which was then gradually articulated. An iterative cycle involved the designer specifying a new input point on the surface of the freeform geometry, which caused the Delaunay triangulation to regenerate (Figs. 120 - 125). This process looped 10 times, at which point the triangulation was automatically exported to a GWA file which was analysed in GSA™ for structural feedback. While the GWA file contained node and member information, it did not contain information about restraints and loading. The analytic process was not automated and had to be performed manually by the designer. In this case, analyses for stress and translation were performed (Fig. 126). Having analysed the structure, the designer then continued with the triangulation process within Rhino™ or chose to stop the process. In resolving this project, it was found that the Delaunay triangulation did indeed give a result that was closer to the freeform surface than a standardised triangulation, and that this result could be achieved with fewer triangles and therefore fewer panels and members (Figs. 127 - 131).

PROJECT SUMMARY
The outcome of this project was knowledge of the geometric input requirements for GSA™ and how to automatically extract this information from the CAD software Rhino™. More importantly, it allowed me to understand the process of constructing a model for structural analysis. Sitting beside the engineer as he applied restraints and loading and ran the structural analysis provided insight into the substantial amount of non-geometric information that is required in addition to that which the architect generally defines in CAD. It also demonstrated the relative ‘clunkiness’ of adding this information, and selecting nodes and members, within GSA™.
Figure 126. Colour plots showing three iterations of analysis for stress and translation
Figure 127. Deviation of Delaunay triangulation from original freeform surface

Figure 128. Deviation of equal triangulation from original freeform surface

Figure 129. Panel detail from final geometric model
Figure 130. Final geometric model - render 1

Figure 131. Final geometric model - render 2
SYNOPSIS

The three projects that I have reported in this section demonstrate the use of 3D digital tools to enable interdependencies between aspects of architectural and engineering design. The processes developed within these projects respond to Holgate’s criticism that “the input offered [by engineers] is unnecessarily accurate, and too slow; to be useful at the early stages of the process” (Holgate 1991: 148), and examine the particular problem of modeling and communicating loosely defined or incomplete design information during the early design process.

Within the three projects, I have used generative and flexible modelling techniques to synthesise architectural and structural design parameters around the common (and ‘low resolution’) language of geometry. Within the Bendigo Canopy project, I developed a tool that synthesised an architectural idea with a set of structural ‘rules of thumb’. This tool supported a more complete appreciation of the design goals of both architects and engineers and, as a generative tool, facilitated quick design iteration and exploration. My exploration of the potentials of this synthetic approach to architectural and engineering design was then extended at a higher level of resolution using CATIA™. The Marina Bay Bridge project explored one practical benefit of greater interdependency between architects and engineers. Through my development of a 3D parametric model, it was possible to answer a highly detailed question about the bridge design at a time when the basic design information which would traditionally support such an exploration was lacking. This illustrated the benefit of 3D digital tools in not just aiding the modeling process but in analysing that geometry to answer questions. Lastly, the School Canopy project, while in essence a device to increase my understanding of the software used by structural engineers, initialised an exploration into transferring geometric design information between CAD and analytical software. This last area of research will be further explored in the following section.
INTRODUCTION
This section investigates the idea that analytic tools might drive aspects of design exploration. In describing the relationship between design and analysis, Kolarevic states that “the current simulation tools are completely useless from a design perspective. If you agree with that as a position, then you ask what will make them useful” (Kolarevic, in Kolarevic 2005: 234). An understanding of usefulness was developed in Chapter Five where, rather than reactively (in)validating a given design, analytic tools were used as an active guide to the design exploration process. This synthesis oriented approach to joint and creative problem solving was defined as performance-based design, a methodology in which architects and engineers are design co-authors.

Within Chapter Five, I raised several issues as significant and problematic in the generative use of analytical software. The first of these concerned differing perceptions and the problems of translation and interpretation. At the most basic level there is a problem of getting data between CAD and simulation tools. Meanings and file-types are often not shared, and considerable rework, ‘cleaning’ and the addition of non-geometric information is required. The outputs of analysis facilitate colour plots, but not CAD-based implementations. If what March and Simon have observed is true, that “the greater the efficiency of communication… the greater the tolerance for interdependence” (March & Simon 1958: 162), then other mechanisms for establishing common meanings need to be developed. Additionally, any such processes of constructive negotiation need to occur between the relatively high resolution nature of the simulation tools and the low resolution nature of the early design process. To explore these issues, this section reports on three projects that use computational performance simulation as an active, generative mechanism within early design exploration.
6.2.1 VENICE BRIDGE
PROJECT DESCRIPTION

This project, Venice Bridge (2006), explores the use of daylight analysis as a generative mechanism and active partner within the design process. It inverts a number of the Bendigo Canopy contexts. While in that study structural ‘rules of thumb’ or analysis were synthesised within a common software program, this project examines the problem of linking together single domain tools. The Bendigo Canopy project was a ‘live’ project, in which Arup interacted with external partners. In contrast, this project was conducted as in-house research, the aim of which was to explore the connection of CAD tools and daylight analysis software. It took place within the framework of an international design competition. This competition was the first that the Arup Melbourne office had entered under its own auspices, and provided a vehicle to extend the practice’s understanding of how information can be moved more efficiently between CAD and daylight analysis software. The competition team included the author and two mechanical engineers; John Morgan, a specialist in environmental physics, and Tai Hollingsbee, a specialist consultant in sustainability and building performance.

The competition brief, which called for an inhabited bridge in Venice (Fig.132), contained a complex programmatic mix of gallery spaces, cafeterias, administration areas and workrooms, each requiring differing levels of natural daylight. For a number of these programs, in particular the gallery spaces, controlling the level of daylight entering the space is a critical design issue. As presented here, the investigation picks up at a point subsequent to the design of an unarticulated building envelope and the structural system. The design problem addressed is the generation of optimally sized...
and located window openings in the façade (Figs.133-134). I firstly describe the approach taken to issues in terms of communication and interaction between the software programs, and then in terms of accessing building performance during the early design phase. It then describes the information flow between the 3D model and the daylight simulation tool Radiance via a database, the simulation process within Radiance and then the transfer of information back into the 3D model.

WHERE SHOULD THEY BE?
This project coincided with the initial stages of the interview process that I conducted with senior Arup employees, and the problem it addresses was particularly well described by one of the interviewees. He gave a description of the process undertaken to design a semi-porous skin covering an existing art gallery. The design intent in that project was to create a semi outdoor space with its own microclimate, and one of the important design questions being asked was “what were the optimum properties for the skin in terms of sizing, location and quantity of the holes?” In describing the role that digital tools played in the development and communication of the design information between architect and engineer, the interviewee, a mechanical engineer, stated that “the tools are useful to produce images, but when it comes to things like how a space is actually affected by a design they’re no good… The question the architects were asking was ‘how big do the holes need to be, and where should they be?’ We couldn’t answer that with colour plots”.

The approach generally taken is to use a daylight simulation tool such as Radiance™, an industry standard tool for ray-tracing, to evaluate a design that has a given configuration of holes, then change the design configuration and evaluate again until a good solution is found. In this approach, the simulation is only helpful in evaluating design options but lacks the ability to directly help in finding a good design. An alternative approach, which was proposed at the start of this project, was to design an optimised façade system by generating the location and sizing of openings directly from the results of daylight analysis. This is one aspect of the problem dealt with in this project.

The other aspect emerged from the nature of the early design phase. While the building envelope had been defined, the configuration of program within the envelope was not yet determined. The design team wanted to test different program configurations, preferably without the need for reanalysis. These two aspects required that a particular approach to the relationship between geometric and analytic frameworks be developed.

INFORMATION TRANSFER: EXPORT AS
Existing methods for transferring design information between the CAD software Rhino™ and the simulation software Radiance™ are limited to the transfer of geometry via the IGES file type. This posed several challenges. Firstly, a significant amount of additional material related
information needs to be applied to all geometry within Radiance™. Secondly, there is no means for importing analytical results back into Rhino™. The default output of results in most daylight simulation tools is limited to numeric data and visualisations - renders or false colour plots, either within the software or via programs like ParaView™.

The design team spent a considerable amount of time during the initial stages of the project talking, drawing and sketching (Figs. 135-137), and much of this discussion concentrated on defining an efficient information exchange process. During these discussions, a large amount of the CFD analyst's knowledge was made explicit. Radiance™ is an idiosyncratic program, in that it is driven with a high level of individual customisation. Working closely with the analyst led to an understanding of exactly what information was required, and how to communicate that most effectively (Fig. 138). The outcome of these discussions was to use an excel spreadsheet, referred to as the design file, as an intermediary between the two software packages. The information required for a Radiance™ simulation includes geometry, defined as nodes and polygons, and materiality. After some adjustment to the 3D model, detailed below, a RhinoScript was written to extract this information from the CAD file into the Design File. Layering within Rhino™ was used to assign simple material definitions to geometry. By using strings recognisable to Radiance™ this intelligence could be carried over into the analysis model.

The exported geometry was limited to only that needed for the simulation, a quantity smaller than that present in the entire CAD model (it is important to note that replicating a geometric
Figure 138: Diagram showing the progression of information through software and scripting stages.
model exactly within an analysis program brings no actual advantage and in many cases it is beneficial not to). It was divided into three types of entity: entities to measure, entities to test, and environmental entities.

ENTITIES TO MEASURE
The initial unarticulated building envelope was split into 750 X 750 mm polygons, each a possible site for a window opening (Fig.139). This grain was chosen because it was fine enough to stop the sunlight coming through any particular panel from flooding the interior, and correlated with the size of standardised building components. There were over 1300 polygons, each associated with a ‘glass’ material attribute. To keep track of both the polygons and the information associated with them in the CAD and analytic environments, as well as in the design file, a unique name was given to each polygon which was common across all software environments.

ENTITIES TO TEST
In addition to providing building geometry it was also necessary to provide ‘problem’ geometry – geometry used in the simulation that is required to define the analytical problem. This was a collection of points, in a 1 X 1m grid, +500mm above floor level, which became the points tested in the Radiance™ simulation (Fig.140). A 1 X 1m grid of points was seen as giving a sufficiently accurate result: the more point locators the finer grained the results are, and the longer analysis takes. This point grid was further divided into nine separate zones, each approximately 100 m². These zones represented the possible program locations within the building.
ENVIRONMENTAL ENTITIES

Environmental entities were defined as entities that effected the simulation, an example of which might be overshadowing buildings that blocked out daylight during certain parts of the day. While the site was clear of this kind of building, the building itself was planned over two floors (Fig.141) and the presence of the floor plate stopped light entering the upper floor from reaching the lower floor. The floor plates were associated with a ‘concrete’ material attribute, were exported directly to Radiance™, rather than to the design file, as an IGES file.

RELATING THE FRAMEWORKS

At this stage testing showed that the design file successfully facilitated the flow of information between the CAD and Radiance™. It did not, however, relate them in any meaningful way. This relationship is created through the approach taken to analysis, which is described further in the following section but can be summarised here. The building was analysed as a series of separate zones. By analysing each zone separately (Fig.143), the relationship between that zone and the façade in terms of direct daylight can be established, independent of any other zone. This information is kept in the design file. Each program has a desired amount of daylight, and through the simulation the amount of daylight reaching each zone through the building façade is known.

The design file is used to determine which program occupies which zone. Because the relationship between zones and the façade have been established independently, different programming configurations can be assigned without the need for re-analysis. A VBA script then creates the façade openings as geometric entities within CAD. Using this approach, the design team was able to explore a solution space of possible program configurations, rather than only a single specific case. The size of the solution space can be defined as the total number of possible program configurations within the building. The following sections expand on this method.

RADIANCE SIMULATION

Within the Radiance™ simulation, each test point casts rays towards the ‘sun’ - a collection of combined XYZ positions that describe its course for each hour of each day of one year. This amounts to thousands of rays for any one point. When a ray passes through a façade mesh, a hit is registered for that mesh (Fig.142). The more hits registered by a mesh the more direct sunlight reaches the point being analysed via that façade element. Through this method the relationship, in terms of direct sunlight, between any particular façade element and any particular location within the building can be determined. Any single façade element typically affects multiple zones.
Figure 143. False colour plots showing 'hits' recorded within each zone. Red signifies a greater number of hits, blue a lesser number.
RADIANCE TO DESIGN FILE INFORMATION TRANSFER

The results of these analyses, a cataloguing each façade mesh and the number of hits recorded by it, were collated in the design file (Fig.144). Within the design file, the designer can test different programmatic configurations, in this case 6 programs types between the 9 zones (Fig.145). This combinatorial testing is achieved by the application of a multiplier specific to each program, which relates the amount of daylight reaching that program to its required daylight factor as specified by standards (Fig.146).

INFORMATION TRANSFER: IMPORT AS

These calculations are used to determine the sizing of façade openings and diffusers. The results are filtered through 8 possible window sizes, which range from 100 to 700mm in width and step in 50mm increments. The window height is always 750mm. If the level of sunlight is too great for a particular program’s required daylight factor, or of no benefit, no opening occurs. A VBA script then imports the calculated window width as the basis for automated geometry generation in Rhino™. The script generates on average 1000 correctly sized and located window openings for any tested combination (Figs.144-148).

PROJECT SUMMARY

This project has demonstrated how digital strategies of practice can enable the use of analytic tools as a design generator. The generative process described was successful in quickly informing the design in a manner appropriate to early design exploration. Rather than using daylight analysis to determine building performance for a single program configuration, the process returned information that allowed the designer to explore many alternate optimally oriented configurations. This is a very different interaction to that which currently occurs in practice, and one that suggests the potential role that analysis can play in supporting architect-engineer design exploration.

The use of a 3D digital model, scripting and a spreadsheet facilitated a direct transfer of information between design and analysis software. The efficiency of this transfer was predicated on knowing exactly what information needed to be provided, and in what format. It is significant that some of this information was not part of the building geometry, but rather related to the requirements of simulation and analysis – in this respect, integration of CAD and daylight analysis does not possess the same directness as that of CAD and structural analysis. It is also significant that closing the loop between CAD and analysis software required representations very different to those easily available within the software (false colour plots). This suggests that the representations that are most useful in facilitating understanding may not be the same as those that promote interoperability, and that there is the need for both.

While a shared knowledge of input and output requirements facilitated efficient communication, designing the process of synthesis was the primary means for facilitating shared authorship. This
Figure 144. Specifying different programmatic configurations within the design file.

Figure 145. Diagram showing possible configurations.

Figure 146. Program sensitivity to direct sunlight.

Figure 147. 4 examples of differing window configurations generated after specifying different program configurations.
required inter-domain problem solving between the architect and the engineer; a collaborative process of constructing a shared understanding and definition of the problem. This discussion represented new territory for me as ‘the architect’ and for the engineers: it is not typical for one to involve the other in any significant depth in the design of their design process. This suggests that in processes of constructive negotiation, the integration of architectural and engineering workflows is as significant an issue as data integration. Because of this, the process resists becoming a black box.
ENVIRONMENT: The project design is driven by daylight requirements of open public spaces with no supportive structure. The fabric is a flexible, lightweight, and transparent material that allows for natural light to penetrate the space. The design incorporates high-performance glass and a design that maximizes the use of natural light.

Light penetration through the façade is achieved by using transparent glass and high-performance glass that allows for natural light to enter the space. The size and shape of the glass panels are determined by the environmental conditions and the specific requirements of each program.

During the daytime hours, the building is designed to provide natural light and actively responds to wind pressures. The design is based on a combination of transparent and semi-transparent materials, which create a sense of openness and transparency. The building is designed to be flexible and adaptable, allowing for changes in lighting conditions.

FORM: The program and façade are integrated within a series of spaces, each with its own characteristics. The design is driven by the requirement for open public spaces, which are designed to maximize the use of natural light and provide a sense of openness and transparency. The design incorporates high-performance glass and a design that maximizes the use of natural light. Throughout the building, each program is integrated into the space, with each space providing distinct and identifiable visual cues to the city, the water, and back onto the bridge.
6.2.2 CHEESE TOWER
PROJECT DESCRIPTION

This project, Cheese Tower (2007), was undertaken as in-house research at Arup Melbourne. It emerged from the desire to extend some of the knowledge gained through the Venice Bridge project, and the same team of Tai Hollingsbee, Jon Morgan and I collaborated in its development. Applying the research generated in the Venice Bridge project to a more generic design situation, the project explores the role that day-lighting analysis can play as a driver of building form.

Large commercial buildings, especially shopping centres, typically demand high levels of environmental comfort and require high levels of energy consumption to provide it. As these buildings are often very deep, atria and light-wells can be important means by which the designer can meet these requirements via passive means and thereby reduce energy consumption (Fig.149).

While atria have the potential to significantly increase energy savings, there is a corresponding negative economic impact: every square meter of atrium space is a square meter of lettable floor-space lost. The amount of daylight entering a building via an atrium depends on a complex set of factors, including orientation and global position, the surrounding context, the depth of the building and the number of stories.

There is currently little computer-based guidance available to support early design decisions regarding atria and light-wells, in particular regarding their size and shape. This project details the development of a tool that, at the massing model phase of a project, determines the most efficient configuration of
light-wells and floor space for a given building mass to allow the highest average daylight factor, evenly distributed, within that building (Fig.150).

BUILDING DEFINITION
Massing models are simplistic representations of building designs, typically representing only the building envelope and floor-plates. They are particularly useful in the early design stages where they support the exploration of building form. In this project, the massing model was a parametrically defined 3D model of a four storey building that included floor plates, a lift core and external walls (Fig.151). The model was built in Generative Components™ (GC), a parametric plug-in to Bentley Microstation™ (Figs.152-153).

Drawing from the knowledge gained in the Venice Bridge project, the geometry within this model was defined so as to be compatible with the delighting simulation process within Radiance™. Each floor plate was divided into quads on a 5 X 5m grid, and a grid of testing points was generated for each floor. In addition to the building geometry, contextual geometry was included within the model. This took the form of a lift core and several overshadowing buildings.

DESIGN FILE
Similarly to the Venice Bridge project, where its use is described in more detail, this project developed a database to pass information between GC™ and Radiance™. The database, an MS Excel™ file, was populated by writing a custom script within GC™ to export the geometry in a form readable by Radiance™. A
second script imported information back into GC™ subsequent to analysis.

RADIANCE SIMULATION
As with Venice Bridge, the core ‘design’ activity was the collaborative development of an approach to analysis. The basic framework for this approach was that, beginning with the massing model as defined by the designer, an iterative process of simulation and alteration would occur within Radiance™, the result of which would be a range of different average daylight factor options output into CAD.

The design team began by testing subtractive and additive approaches: in the subtractive approach, the building was iteratively ‘eaten away’; while in the additive approach it was gradually built up. We then explored using solid or transparent floor-plate materials and whether or not to constrain ‘growth’ to a boundary condition (Figs.154-155). The most effective of these was found to be a variation of the additive approach, in which each floor quad was treated as an entity whose transparency could vary.

This process (Figs.156-157) begins with the materiality of all floor-plate quads defined as completely transparent. After taking an initial measurement of the daylight factor in the scene, one by one each floor quad is set to 100% opaque, the impact of this change simulated and the material set back to its previous transparency. Having looped through all quads, the total list of quads was sorted according to their impact on the daylight factor. The opacity of all quads was then increased, with the quad recording least impact on the overall daylight factor increasing its opacity by 10% and the quad recording most impact increasing by 1%. A veto function prevented material change if that change led to the daylight factor moving below a minimum, which was set at 2.5. This process iterated until the whole building was filled in (Figs.162-163). When significant changes in the minimum daylight factor were recorded, for example when it reached 10, 5, 4, 3, 2.5 etc, the state of the model was recorded to the design file.

INTEGRATION & EXTENSION
A script was written within GC™ to reconnect the 3D model with the design file. As described previously, the state of the model was recorded when significant states benchmarks were passed, being average daylight factors of 10, 5, 4, 3 and 2.5. This involved recording, within the design file, which floor quads had a transparency of 40% or less when those daylight factors were reached. By using that information as an input into the 3D model the designer, using a scroll bar within GC™ (Fig.158), was able to see the configuration of the model at each of those states (Figs.159-161).
Figure 154. Sequence describing an iterative, additive process whereby the definition of new floor quad elements is constrained to the edges of existing floor quad elements.

Figure 155. Diagram of the above described sequence.
Figure 156. Diagram showing final analysis/synthesis process

Figure 157. Diagram showing iterative opacity process - 'greyer' quads are more opaque
Figure 158. False colour plots showing daylight factor measured over 10 iterations
Figure 139. Colour plots showing iterative increase in opacity over 10 iterations.
Re-importing analytic data back into the GC model - basic model with column grid

Re-importing analytic data back into the GC model - testing a minimum DF of 2%

Re-importing analytic data back into the GC model - testing a minimum DF of 5%

Using the GC 'slider' to test different minimum daylight factors
A SUBSEQUENT APPROACH

A significant limiting factor in the above approach was the speed with which a solution could be found. Within the test project, the exhaustive nature of the process meant that one analysis was performed for each of approximately 300 quads per iteration, for typically 20 iterations. This totalled approximately 600 analyses per run, taking approximately eight hours using a computer with standard specifications.

To facilitate faster exploration, an approach based upon simulated annealing was developed. The analysis software was Ecotect™ (rather than Radiance™), a general purpose software for early design simulation and exploration. One benefit of using Ecotect™ was that the geometric and analytic aspects of this project could both occur within the same program, eliminating the need to bridge between design and analytic softwares using the design file.

An optimisation approach based on simulated annealing seeks to find acceptably good solutions, and although it “is unlikely to find the optimum solution, it can often find a very good solution” (Carr 2008: n.p.). Using a simulated annealing algorithm allowed for a reduction in the number of analyses performed to one per iteration with, in application, typically 200 iterations per optimisation run. The objective function used is Daylight Factor, which is measured as an average recorded by the Ecotect™ analysis grid. One constraint is placed upon this function, that the number of quads assigned to a glass material must equal the total number of quads * 0.2 (e.g. the formal optimisation model is max(f), s.t. g = h*0.2, where f = Daylight Factor, g = the number of glass quads and h = the total number of quads). The simulated annealing algorithm was coded in Lua, the endemic scripting language of Ecotect™, the pseudo-code for which is as follows:

StateCurrent = s; EnergyInit = DF(StateCurrent); EnergyBest = 0
While Counter < CounterMax
    generate a neighbouring state (NewState) from StateCurrent
    calculate the Daylight Factor for NewState, EnergyNew
    determine difference in energy between EnergyInit and EnergyNew, EnergyDif
    if (energyDif > energyBest) then accept new state as StateCurrent and EnergyNew as EnergyBest
    else if (Rnd < exp(-energyDif / T)) then accept a non improving move
    Else reset to StateBest
End Loop
Return StateBest

where S represents an initial starting state, and Temperature (T) is determined via a schedule, and lowers as the search progresses. The schedule is mathematically related to the number of iterations (240), and occupies the range of 5 to 0.5. A new solution is accepted if EnergyDif > EnergyBest. If EnergyDif < EnergyBest, the new solution is accepted using the Boltzman distribution, e.g. with the probability of exp(-EnergyDif/T). Fig 169 describes the the decreasing likelihood of non-improving solutions being accepted using this acceptance probability function, as well as the progressive maximisation of the objective function over 240 iterations. A

---

7 Ecotect is developed by Dr Andrew Marsh. Further information about this software is available at http://www.squ1.com/index.html
random function is used to restart the algorithm from the best recorded solution and to restart from a random state. The full script can be found in Appendix A4.2.

Similarly to the previous exhaustive approach, the analysis performed within Ecotect™ calculates the average daylight factor recorded on the analysis grid, which is offset +500mm from the floor quads (Fig.164) at each level (Figs.165-167). Design and non-design geometry can be specified and assigned material properties within Ecotect™. The daylight factor is calculated against the position of the sun at each hour of each day for a year (Fig.168). In contrast to the previous approach, floor quads are assigned a concrete or a glass material property rather than progressing through a range of opacities. Neighbouring solutions are generated from the current solution, with a glass configuration that is 50% identical to the current solution. 50% of the current glass configuration remains as glass, while the other 50% is swapped randomly with cells that are specified as concrete within the current solution. Typically, 200 potential solutions were generated in each run (Fig.169).

Several tests were conducted with differing calculation precision values within Ecotect™ (Fig.170). Calculation values from 1 (lowest/quickest) to 4 (highest/slowest) determine the number of test rays that are projected from each test point on the analysis grid. Using a precision of 4, an optimisation process limited to 200 iterations took less than 2 hours, while using precision 2 took approximately 7 hours. There do not appear to be any marked differences in the results, except for perhaps a greater level of ‘connectedness’ exhibited in the distribution of openings in precision 4.
Figure 168. Diagram showing shadows cast by the sun at the site over the period of a year.

Figure 169. Diagram showing average DF result over 200 iterations of the simulated annealing process.
results. A precision of 3 was used for all subsequent testing.

Changing the configuration of surrounding buildings, and adding elements such as lift cores (Figs.171-172) led to differing results which reflected those conditions. Testing these solutions against a ‘standard’ atrium shape in the same conditions (Figs.173-174), with both solutions sharing the same area of opening, revealed that the average daylight factor throughout the optimally directed solution was substantially higher than that of the standard configuration.

![Image of simulation results](image)

*Figure 170. Testing simulation results of different calculation precisions within Ecotect*

**PROJECT SUMMARY**

This project successfully demonstrated the use of 3D digital tools to support a process of early design exploration that integrates lighting analysis and architectural design. While this tool was successful in generating optimally directed atrium designs, several problems and bugs were encountered. Foremost amongst these was that, using the exhaustive method, obviously incorrect results were generated when buildings had more than four levels. This problem was initially ascribed to the number of times light was allowed to ‘bounce’ within the Radiance™ scene, which was set to 3, however either increasing or decreasing this setting continued to generate obviously incorrect results. The solution to this problem remained unresolved. Secondly, the linkage between the 3D model and the design file was limited by the idea that the floor was either ‘there’, as a concrete floor, or ‘not there’, as an atrium or void. Because of this distinction an arbitrary figure of 40% transparency or less was used to differentiate between the different floor quads. The later development of the simulated annealing approach made these problems redundant.

When presenting this project to engineers in the Arup Berlin office during 2007, it was remarked that a growing trend in building design was to use transparent glass within the floor-plates of buildings. This activity would bring architects and engineers into close interaction during the early design phase and, as developed, the approach developed within this project would provide support to such designs.
Geometric results of optimisation process - Figure 171. Configuration 1

Geometric results of optimisation process - Figure 172. Configuration 2

Typical configuration - average daylight factor throughout building: 2.24649

Optimised configuration - average daylight factor throughout building: 3.24362
6.2.3 SKYBRIDGE.
PROJECT DESCRIPTION
This project, SkyBridge (2005), is for a footbridge, connecting two buildings over an alley with the pedestrian nature of that alley requiring that the footbridge be supported from the buildings. I conducted the project as in-house research to gain familiarity with GSA™, Arup’s structural analysis software. The project explores the use of structural analysis as a means to find form, or to suggest possibilities rather than produce optimal structural solutions. In this sense, it is in the tradition of Gaudí and Otto’s analogue form finding processes.

THE MODEL
A simple 3D model was developed using Rhino™ (Fig.175). It included two types of point entities that are defined by the designer. The first types of points represent loads, and within the model traced the pedestrian path across the bridge. The second type of points represent restraints, each being a point at which the bridge is supported by the walls on either side of the alley.

To generate a random starting point, a RhinoScript was used to fill the space of the alley with 100 randomly positioned points (Fig.176). The script then connected each point to its nearest five points, resulting in a ‘mess’ of connected steel members.

THE ANALYSIS LOOP
GSA™ was run automatically from MS Excel™ using VBA™ (Fig.177). Each iteration of the process consists of several steps. On the first iteration, node and member information was
automatically extracted from the Rhino™ model, and formatted as a GWA file, essentially an MS Excel™ spreadsheet (subsequent iterations refer to the GWA file). As part of the extraction process, additional information regarding the size of loads, member properties and other structural aspects required within GSA™ was applied. This file was then analysed within GSA™, and the results for displacement at each node returned to the MS Excel™ spreadsheet. Using these figures the nodes were ranked from highest to lowest, and the nodes recording the lowest displacement (those not working hard) were moved towards the highest (those working hardest). As the model converges on an efficient configuration, superimposed nodes are deleted and the size of that particular member increased. Each iteration ran as a single step, the stopping point being defined by the designer’s choice to stop the process.

This process was tested on a configuration of four restraint points and a single central load, where the resulting form closely approximated an arch (Fig.178). When the process is applied to the alley site (Fig.179), with six points of restraint (the points at which the structure could connect to adjoining buildings) and four load points (the points at which the bridge deck connected to the structure), the resulting space truss structure demonstrates how a form might be generated by combining “the architectural intention and formal design freedom with engineering creativity in regard to the rules of stress flow” (Kloft 2006: 90). (Figs.180-182)

PROJECT SUMMARY
Though simple in its scope, this project demonstrates the potential of making structural analysis an active partner in the design process. The project led to an enlarged solution-space for the designer, informed by structural analysis, which could be explored at speed and with relative ease. There are many potential directions in which the approach could be extended; some of these would include a mechanism for comparing alternative designs, and to provide greater control over the properties of the elements – for example to explore tension and compression elements. Another potential approach would be to allow the means by which the designer currently controls the process, being the load and restraint points, to be the elements that are free to change.

SYNOPSIS
The three projects that I have reported in this section demonstrate the use of 3D digital tools to enable interdependencies between aspects of design and analysis. Rather than replicating the traditional reactive use of analytical software, within these projects I have employed a performance-based design approach that has deployed analytical tools as active drivers in the process of design generation.

Each of the three reported projects ‘closed the loop’ between CAD and analytic software in ways consistent with the process of early design exploration. Within the Venice Bridge project, the digital process returned analytical information that allowed the design team to explore different programmatic configurations, and their corresponding façade opening configurations, within

8 No other structural constraints were considered in this process.
CAD software. In the Cheese Tower project, I developed a tool that informed the placement of light-wells and atria within building floor plates at a ‘mass model’ stage of design. The Skybridge project, which explored structural rather than environmental analysis, used stress flow as a means to combine architectural and engineering design interests. All three projects involved inter-domain problem solving between the architect and the engineer; a collaborative process of constructing a shared understanding and definition of particular design problems. In each case, software interoperability and interdependency was reliant on an increased understanding of the workflow and intentions of the other.
Figure 178. Iterative form finding process - test run with single central load and four restraints
Figure 179. Iterative form finding process - pedestrian bridge with multiple loads and restraints
Figure 180. Form-found pedestrian bridge - perspective

Figure 181. Form-found pedestrian bridge - elevation

Figure 182. Form-found pedestrian bridge - perspective plan
INTRODUCTION

The previous sections have explored how digital models that engage different aspects of building performance can support integrated design exploration, firstly by operating in situations of low resolution and secondly by enabling the active use of simulation as a design driver. These studies have shown that the digital practices which support collaboration between architects and engineers can develop around different design processes and support different information flows, but that their common characteristic is that they enable interdependency and the co-presence of architectural and engineering design thinking.

This section extends that research by investigating use of the 3D digital medium as a tool for procuring detailed design information about fabrication. Szalapaj has noted that “the secret, if there is such a thing, with digital tools in practice, lies in the way they support connections between the design of complex sculptural forms and the rational methods of fabrication and construction that are needed to realise them” (Szalapaj 2005: 10). This statement opens up the question of what kind of connections might be required, a better understanding of which is important since connections often require some work to achieve – they go against the grain of more typical construction experience where “consultants fiercely maintain their independence, contractors compete for work and specialists struggle to maintain the integrity of their skills against market driven demands for lower costs and faster delivery” (Bennett & Peace 2006: 7).

The interview process has provided some insight into the role ‘making’ might play in architect engineer interaction: Mr. A stated that “there are benefits for engineers and certainly for architects to really understand how things go together, particularly complex forms, and how the initial design assumptions can be made to work far more efficiently”. An example of that efficiency is the ability to ‘tune’ a design specifically for the fabricator who will make it. Also highlighted, however, was that this input is often missing: “typically, a design is produced and tendered without the input from the people who are going to manufacture and erect it. The expertise of these people, which is quite considerable, is often lost – it’s a fait accompli by the time they are on board” (Mr. A). In this environment, an all too common situation is that “the contractor uses his or her construction ‘knowledge’ once site production has begun to create all sorts of claims to overcome the in-built design inefficiencies” (Cornick & Mather 1999: 121).

As Cook has noted, “the freer people are when they’re thinking about a building, the more important it is to button down into how it’s going to be made. Therefore we’ll have to be working more and more closely with the people who are making it” (Cook, in Castle 2002: 78). What role can 3D digital tools play in supporting such an interaction? To address these issues, this section examines the use of 3D digital tools to facilitate architectural and engineering interdependence and negotiation around processes of fabrication.
6.3.1 THE TRAVELLERS
PROJECT DESCRIPTION

This project is a series of 10 stainless steel sculptural figures, named ‘The Travellers’, designed by Nadim Karam and Atelier Hapsitus and constructed in 2006 on Sandridge Bridge, Melbourne (Fig.183). Each ‘traveller’ is a free standing stainless steel frame figure 7.5 m tall and between 5-10 m wide comprising of hundreds of connecting stainless steel RHS pieces (Fig.184), formed from “families” of members that make up two similar planar surfaces connected by a series of diagonal curves (Fig.185). Each planar member and its identical “twin” in the parallel plane (they average 750mm apart), are joined together by a diagonal member whose curvature in turn is defined by the two co planar members. The process of designing and constructing the Travellers was characterized by complex geometry, a limited budget and a project timescale of 9 months between conceptual sketch and being completed on site.

The project artist, Nadim Karam, was located in Beirut and the rest of the project team in Melbourne. This team included the Melbourne City Council (MCC) as project manager and client, Arup Melbourne structural and mechanical engineers and steel fabricators Silverstone Engineering and DanFab. To respond successfully to the issues of geometric complexity, budget and timescale the Travellers project required a design and procurement process that is unusual in an era of design-bid-build and competitive tendering. A key decision was made to include a specialist stainless steel fabricator as part of the project team from the beginning, so that practical fabrication advice could inform the process from the outset. The following sections describe how the impact of working ‘more closely with the makers’
EARLY COSTING AND MATERIAL PURCHASE

The project began unconventionally with the pre-purchase of materials. The Travellers are made from stainless steel RHS, which is imported into Australia with a lead time of 12 or more weeks. This represented a third of the total project time, and it was quickly realised that it would be necessary to pre-purchase the steel before the design, analysis and fabrication processes had been finalised. Initial spline based 3D Rhino™ models had been supplied by the artist, from which rough schedules and costing information could be extracted based on member lengths. After this information was extracted from the 3D models, the project manager (MCC) pre-purchased the stainless steel required to fabricate the sculptures - a quantity that was equal to all the stainless steel available in Australia at that time. As well as taking advantage of favourable pricing conditions, this removed a number of variables from the subsequent tender process: precise material lists (of which there was now a guaranteed supply) were supplied as part of the tender documents.

FULL SCALE PROTOTYPING

The process of developing a modelling and representation strategy that best fit fabrication and analytical requirements also commenced at the project outset. Scaled wax rapid prototypes (Fig.187) and 1:1 steel prototypes (Fig.188) were generated from the 3D models to explore

(Fig.186) affected the project, and the means by which 3D digital tools were used to integrate the intents and requirement of the designer, engineer and the fabricator.
different connection options, which were discussed with the artist, architect, engineer and fabricator. The prototyping process gave all parties the ability to understand the requirements of the fabrication process and the geometric constraints that needed to be taken into account, and to deal with fine detail at a time when design had only just begun. As well as resolving issues about preferred connections and finishes, the prototyping process was instrumental in identifying where conflicts occurred between analytical and fabrication requirements (Figs.190-192), and in identifying strategies that minimised welding.

The largest and most complex sculpture was developed further than the others and provided the basis for a full scale prototype (Fig.199). At an early stage, it became obvious that it would not be possible to fabricate all the sculptures in the given time without additional geometric rationalisation of the radii used in the curving of the sections, particularly the larger 80X80 perimeter sections. This knowledge, as well as that gained about other geometric, representational and fabrication requirements, informed a geometric rationalisation process that was then applied to all the figures.

GEOMETRIC RATIONALISATION

The first step in the rationalisation process was to approximate the artist's spline geometry with a standard series of arcs and straights (Fig.193). This process was coded as an automated process within Rhino™. The script found the inflexion points along each spline, split the spline at these points and then matched each fragment with the closest arc from a standard series. Arcs were separated by a straight line tangential to each – by inserting this minimum
Figure 193. Geometry rationalisation - curves are defined by arc segments

Figure 194. Comparison between original spline curves and rationalised curves

Figure 195. Steps in rationalisation script
length line each member could be rolled as a continuous piece, greatly minimising time spent welding and polishing in the shop.

A second script generated each diagonal member. Geometrically, each sculpture is defined by two planes and a series of diagonal members that move between them, and these diagonal members can be generated directly from the corresponding members on the top and bottom planes. Automating the rationalisation process greatly decreased the time needed to model each sculpture, and resulted in a standardised set of parts that significantly reduced the time required for fabrication but closely approximated the original geometry (Fig.194). Additionally, the process ensures accuracy and nodal connectivity between members, streamlining the analysis process. This process is diagrammed in Fig.195.

**STRUCTURAL ANALYSIS**

Because of the pre-purchase of steel, structural analysis for 9 of the 10 sculptures was performed after the stainless steel sections had already been acquired (Figs.196-197). 316 L stainless steel RHS was chosen as the primary material, and a sufficient degree of triangulation was incorporated at the nodes of steel member to member connections to enable for a relatively efficient structural system to be developed. Full strength butt welding at joints provide rigidity at all joints and also contributed significantly to the stiffness of the system.

Central to the structural design development process was the management of a single electronic file for each figure. This file was circulated in turn to the client, architect, and Arup designers, to be worked on/developed
at each interface. Arup firstly rationalized the proposed geometry, and then analysed and checked these models for structural adequacy. Changes were proposed as necessary to ensure the sculptures were both buildable and structurally adequate, and these would be fed back to the client and ultimately the artist for their consideration. Convergence to a solution typically required two or three iterations to this process, and the City of Melbourne played an important role as reviewer and in facilitating the speedy flow of information to and from the artist.

The Arup in house structural software GSA™ was used during the analysis, with input directly from Rhino™. The 3D model was prepared prior to importation into GSA™ so that all arcs were pre-faceted to control their approximation and to ensure that the geometric file was as ‘ready for analysis’ as possible. It was important to model the geometry accurately to better estimate the structural performance of each Traveller. The models were exported as DXF files; later design iterations and refinement generally involved tweaking, deleting, and inserting individual members, and not re-exporting the entire model.

BUILDING WITHOUT DRAWING
A series of protocols were developed to coordinate the modelling process between the various parties, which codified the ‘rules of engagement’. These included a specified information flow and required that information passing between the parties would be 3D models in specific formats. The digital models contained only centreline geometry and numeric annotations and, along with a series of spreadsheets, were the only documentation produced.
Documentation for each sculpture consisted of a 3D centreline model and an 8 page spreadsheet, one page per section type (Fig.201). A visual basic routine run from Excel™ extracted length and radius information from each member to the spreadsheet, and tagged each member and element numerically within the 3D model (Fig.203). The time taken to extract the spreadsheets was less than half an hour per sculpture, and the information they contained was fed directly into the bending machinery. The 3D model was used by the fabricators to check the numerical spreadsheet information on the shop floor. To aid the process of moving from spreadsheet to model, a scripted routine was written to select and zoom to any given member within the 3D model from the spreadsheet (Fig.204), as it was found that it was easier to locate members from the spreadsheet than to search for them within the 3D model.

MANUFACTURING PROCESS

Digital technologies were used extensively in getting 4455 stainless steel pieces onto the shop floor (Figs.198-199), the figures needed to be assembled, welded and polished by hand. While future projects seem certain to make use of emerging Radio Frequency Identification (RFID)
tagging technology to help automate parts of this process, in this case a hybrid strategy of digital data and templating was employed. The digital extraction process numbered each piece, and this mechanism, in conjunction with plotting a full scale centreline template for each figure (Figs.200 & 202), assisted the fabricators to locate the members in each sculpture.

Close and early involvement from the fabricator, and the production of a full scale prototype, had a significant impact on the manufacturing process. The manufacturer of the prototype effectively ran workshops for the other three fabricators involved, going through the process and lessons learnt with each as the prototype was being built. This gave all fabricators, as well as the extended design team, confidence that the project could be systematically approached despite its apparent complexity, and again represents the highly unusual level of cooperation and information sharing achieved in this project.

The experience of ‘The Travellers’ suggests that the success of this project might be scaled up and replicated on larger projects. An important factor to consider in speculating upon how this might occur is that the Travellers was characterised by the use of a single material: steel. Larger projects, for example a stadium or bridge, would include other materials, an obvious example being cladding materials.

Reflecting upon the processes employed on ‘The Travellers, the introduction of other materials, and therefore fabricators, would involve enlarging the process of negotiation needed to develop a modeling and representation strategy that fit the demands of the different fabrication deployments. In the same way that fabrication and structural analysis requirements were adapted to one another in the ‘Travellers’, resolving the interfaces between the informational needs of multiple fabricators would require adaptation to the standard representation of information, perhaps to a more complicated level. Multiple means for reporting and representing fabrication information would need to be developed (one for each material/fabricator), increasing the importance of a cross-checking process. As occurred within ‘The Travellers’, an early 1:1 prototyping process would be an efficient way to achieve this.

PROJECT SUMMARY

In his recent review of The Travellers project, ‘Something Rich and Strange’, Ronald Jones discusses the divide between design and production, warning that “the disjunction between conception and execution means artists miss vital iterative relationships between idea, medium and technique” (Jones 2006: 30, Appendix A2). While traditional notions of authorship fixate on conception and relegate production to “the labour of drones” (2006: 30), in this case the collaboration between drones is credited with the successful translation from concept to a well built “pearl” (2006: 30).

Implicit and significant to this process were the level of engagement by the fabricator early in the piece, and the spirit of cooperation displayed by all disciplines to working towards favourable outcomes for the client and ultimately for all involved. A comparison with the traditional delivery
process of jobs within the building industry reveals the following:

Very few Requests for Information (RFI's) were generated during the fabrication process, as design and buildability issues were generally resolved during the design process with the contribution of the fabricator. Significantly more information was available at the tender stage than is usual, reducing the perception of complexity and associated costs.

As information describing the full set out of each Traveller was passed on to the fabricator in an electronic format, shop drawings were not required at all. This significantly streamlined the fabrication process, as there was no requirement for the information to be reinterpreted, “handled” and checked a second time prior to enabling for fabrication. The absence of shop drawings meant that real savings were made in time, money and resources - while an estimated $20-30K was saved in fees for the production, review and finalisation of shop drawings, the real savings occurred in the avoidance of delay and the associated risk of liquidated damages.

The culture of cooperation consequently allowed for the efficient production and supply of information from the design team to the fabricators. This meant that the often confrontational and sometimes discordant nature of builder / fabricator /design team interplay frequently observed on many building projects did not beset the delivery of the Travellers project.

In addressing the benefit that digital tools brought to the process of realizing the Travellers, we can immediately move beyond the low hanging fruit of increased efficiency and the ‘file to factory’ communication of complex forms. Rather, the point is that the entire design and fabrication team were involved in processes of learning, and if the same project had moved through a traditional representation and procurement cycle then both that learning, and the collaboration engendered, would not have occurred. In taking an unconventional approach to procurement, supported by 3D digital tools the entire design and construction team were able to increase their own involvement and that of others, transgressing many traditional boundaries.
Figure 205. Installation of a sculpture on site

Figure 206. The Travellers on site 1

Figure 207. The Travellers on site 2

Figure 208. The Technoman sculpture - detail
6.3.2 RECTANGULAR PITCH STADIUM
PROJECT DESCRIPTION

This project, Rectangular Pitch Stadium (2005-2006), is a stadium designed by COX architects. My introduction to the project occurred when I was ‘invited’ to the COX office a few days before an important submission to do some emergency ‘sculpting’ of the stadium form. The geometry provided by Arup (Fig.209), specifically the falloff in the height of the bays as they progressed from the centre to the edges of the stadium, did not match the architect’s intention. There was no easy way of adjusting the engineers’ model within a few hours to meet this intention, so instead I produced a ‘fudge’ that was good enough for that stage of the project. This problem of matching the preferred structural and sculptural aspects of the stadium would remain with the project and, in the context of fabrication and rationalisation, trigger a number of investigations into how structural and formal intentions might be integrated.

CATIA MODEL - ELLIPSOIDS

In the initial design, the geometry of the bay shells was defined as a series of cut ellipses, with each group of three forming a bay (Figs.210-211). Arcs running in a horizontal direction between each ellipse formed construction geometry from which a triangulated structural system was developed. This use of geometry was extremely effective in communicating the project, but brought with it the expectation that standard geometry meant standardised parts. Of course, making arcs between three differently sized ellipses does not generate a standard geometric surface.

To understand the problem, some simple RhinoScripts were written that analysed the
model ‘as is’. These scripts explored two potential paths to rationalisation – one that focussed on panels and the other on nodes. The first script coloured each panel according to its deviation from the average surface area (Fig.212). The second measured the angular deviation of each structural member at each nodal point (Fig.213-214). These studies showed, in an easily communicable graphic form, that while there was variation between the panels and the angles coming into each node, it was not that great, and that there might be a number of geometric strategies to deal with this. The perceived costs of fabricating a large number of different panels were judged as being greater than different nodes, and so an exploration into geometric rationalisation began.

A 3D model was built within CATIA™ to explore replacing the ellipse-based geometry with ellipsoidal geometry (Figs.215-216). Using ellipsoids as an underlying geometry, it was possible to generate symmetry within each bay and ‘bands’ of related panels (Fig.217). The model was constructed using the same rules relating to sightlines and boundaries that had informed the initial design. By adjusting the parameters, it was possible to match closely the original geometry and record a reduction in the number of unique panels by 25%, from 600 to 450. This option was discussed with the architects, however in their view did not fit the initial model closely enough (Fig.218).

PANEL RATIONALISATION

Once it had been realised that the overall shape of the stadium was not for changing, a different approach to rationalisation was sought. This focused on the framing system. If a framing
system with a flexible gasket was used, it would provide a certain amount of tolerance that would allow different shaped panels to fit into the same frame. Panels that fit within a given frame could then be made identical.

While the geometric information to explore this approach was available within the 3D model, it was very difficult to extract, analyse and make use of. There were 600 different panels, each which needed to be compared against all others. Comparing the surface area of each panel, often a fast way of finding matches, was not enough, as panels with the same area could have very different shapes. By hand, accomplishing the task would involve moving, copying and orienting each panel onto a potential frame and then rotating it until it fit or otherwise (Fig. 219) – an unaffordably lengthy process.

Instead a RhinoScript which simply automated the steps outlined above was written to interrogate the 3D model and identify panels that fit within common frames (Figs. 220-228). This resulted in a significant reduction in the number of different panels; at this point however, it was decided that the costs of fabricating different panels would not be much more than fabricating rationalised ones. Rather, the costs of fabricating different nodes demanded that they should be rationalised.
The investigation showed that rationalisation of panel shapes could reduce the number of unique panels by 40%, and therefore reduce cost, but there was no link between this investigation and the change in cost priorities. This change was a response to new information about current fabrication and material sourcing and costing, which suggested that the design team’s assumption that non-rationalised panels would incur a significant cost penalty was not the case. Although contrary to the initial design assumptions, this situation typifies the impact that new information can have upon entering the design process, reinforcing the importance of maximising the input of construction expertise.

PROJECT SUMMARY
This project has described the use of 3D digital tools to explore issues of rationalisation and the integration of form and structure. These investigations in one sense ended in failure. Construction however, more than any other part of the process, is driven by cost. The risks are so large – and often unfairly shared – that to not do so means potential commercial failure. In this sense, this project has successfully demonstrated how the use of 3D digital tools can facilitate a better accessibility to information about potential costs, as well as be used to undertake relatively sophisticated investigations into potential rationalisation strategies. The success or failure of these investigations, however, has been demonstrated to depend at least partly on factors that are not technical, and in this instance related to perceived cost and the desire to maintain a particular shape.

SYNOPSIS
The two projects that I have reported in this section demonstrate the use of 3D digital tools to enable interdependencies between aspects of design and fabrication. Within each project, I have applied 3D digital tools to procure design information which traditionally would not be available during the early stages of design. This information has been used to guide the design process, increasing the potential deliverability of the projects by increasing understanding and, in ‘The Travellers’ project, demonstrably decreasing time and cost.

In ‘The Travellers’ project, early interaction with the fabricator led to digital strategies that ‘tuned’ the process of design and the communication of design information. The project would not have been realised on time nor to budget had a traditional process of shop drawing been followed. In the ‘Rectangular Pitch Stadium’ project, 3D digital tools were used to analyse and increase the understanding of the geometrical and panel design. This analysis led to a series of pre-emptive rationalisation studies which, quickly and to a depth that provided useful guidance, informed the architectural and engineering design discussion with regard to cost implications inherent in the design.
Bay 1 - panels coloured according to surface area

Bay 2 - panels coloured according to surface area

Bay 3 - panels coloured according to surface area

Bay 4 - panels coloured according to surface area

Bay 5 - panels coloured according to surface area

Bay 6 - panels coloured according to surface area
Figure 226. Identifying panel groupings that could be rationalised

Figure 227. Panel groupings shown on shells
SUMMARY
This chapter has detailed projects that I have undertaken within the Arup Melbourne office from 2005 to 2007. The projects have differed in nature: some have been undertaken entirely ‘live’ and involved interaction with external parties and response to deadlines, costing and other project pressures. Others have been undertaken as in-house research, exploring linkages within the multi-disciplinary Arup Melbourne office. A third category of projects are a mixture of both ‘live’ and in-house research oriented activity.

As a group, the projects provide a practical substantiation of the central claim of this thesis, that 3D digital tools can enable the intersection of architectural and engineering early design exploration, and have done so within the context and resources of practice. Each project has used 3D digital tools to facilitate interdependency within architect engineer design exploration, and this has produced project outcomes that extend that which either discipline could achieve alone. The implications and findings from these projects will be discussed within the following chapter.
Chapter Seven:
Discussion and Conclusions

7.1 CONCEPTUAL PARADIGMS
7.2 PRACTICE-BASED OUTCOMES
INTRODUCTION
This thesis has documented my strategies to explore the extent to which the 3D digital environment might offer different modes of interaction, and potentially new forms of collaboration, between architectural and engineering designers. The primary vehicle for investigation has been practice-based project work, and the principal contribution of this research is to explore this problem from within the context, conditions and pressures of live practice. Within this section I discuss the implications of the practice-based projects and summarise the key result of this research: that *3D digital tools can enable interdependencies between crucial aspects of architectural and engineering design exploration during the early design phase.*
7.1 CONCEPTUAL PARADIGMS

At the core of this research is the idea of working constructively across the architect-engineer boundary. To improve this constructive relationship, I have pursued the concept of interdependency. To delineate this concept, I have developed a conceptual framework that draws on organisation theory, design literature and the experiences and understandings of nine senior Arup practitioners whom I have interviewed. To test the application of 3D digital tools in facilitating interdependency, I have engaged in eight practice-based projects within the Melbourne office of Arup.

Within Chapter One, I introduced three broad frames for my research:

- my own motivations and observations,
- RMIT University’s ‘Embedded Research within Architectural Practice’ program, through which I was able to pursue these motivations, and
- Arup, the consulting engineers within whom I undertook the research.

In Chapter Two, I detailed the design of my research. I enlarged upon the contexts of the ‘Embedded Research within Architectural Practice’ program and the Melbourne office of Arup, and then described the two research instruments which I have employed. These instruments were semi-structured interviews, extracts from which I have used within Chapters Three and Five, and the undertaking of practice-based projects within Arup Melbourne.

In Chapter Three, I examined selected design literature to detail several factors impacting upon the historic and contemporary relationship between architects and engineers, and to introduce the problem towards which this thesis is addressed. I described a process of specialisation that has led architects and engineers to see different aspects of a common problem, and introduced the idea that the current sequential approach to interaction was limited by a very restricted interface. In examining the increasing calls by theorists that architects and engineers should interact earlier and more closely, I identified three commonly held propositions:

- interaction should occur during the early design phase,
- interaction should be supported by 3D digital tools, and
- interaction should be facilitated by processes of collaboration.

I then described the complex nature of the early design phase, the until now limited impact of 3D tools during this phase, and the idea that designers find collaboration amongst themselves difficult to control.

Within Chapter Four, I examined the concept of collaboration as a means to initialise my concept of interdependency. I chose to differentiate interdependency from collaboration because of the inconsistent manner in which the latter term is employed. The problematic definition of collaboration within the professional literature may be a factor preventing architects and engineers
from understanding how to work productively across disciplines. I initialised this framework through reference to organisation theory and selected design literature.

Interdependency was defined in Chapter Four as follows:

Interdependency is a productive form of practice enabled by mutual and lateral dependence. Interdependent parties use problem solving processes that meet not only their own respective goals, but also those of others, by constructively engaging difference across their boundaries to actively search for solutions that go beyond the limits of singular domains.

From the literature, I identified four sites of intersection crucial to an understanding of interdependency; these were differing perceptions, shared and creative problem solving, communication and trust.

Within Chapter Five, I grounded these four sites of intersection within contemporary issues of digital architectural and engineering practice. Each site was developed firstly through reference to design literature and secondly through the experiences and understandings of senior practitioners as captured through my interviews. This overlap allowed me to locate particular digital limits to and potential solutions for interdependent design exploration between architects and engineers.

Using a combination of design literature and the grounded experiences of Arup practitioners, I extended:

- the understanding of differing perceptions through reference to problems associated with digital information transfer.
- the understanding of joint and creative problem solving by connecting it to the notion of performance-based design.
- the understanding of communication by focussing it upon the idea of back propagating design information.
- the understanding of trust by connecting it to the management and reduction of perceived complexity and risk.

In concluding the development of my conceptual framework I made four claims about the role of 3D digital tools in facilitating interdependencies between aspects of architectural and engineering design exploration during the early design phase.

I then presented the accounts and results of practice-based projects in Chapter Six, within which I tested the idea of enabling architect engineer interdependency across three spectrums of practice: 1. ‘Design (Arch) | Design (Eng)’, 2. ‘Design | Analysis’ and 3. ‘Design | Making’.

224
TESTING THE CLAIMS IN DETAIL

The four claims made in Chapter Five provide a framework for discussing the research and testing that I have undertaken by practice-based project.

Evidence supporting the first claim, that a pragmatic approach to translation, whereby meaning is attached to operational consequences, can enable interdependent working between architects and engineers within the early design phase, can be found across all projects. These projects confirmed that moving design information efficiently between CAD and analytical software is made difficult by the very limited range of inputs and outputs possessed by CAD and analytical programs. These limitations are primarily geometrical for CAD, and numerical and colour plots for analysis, which often cannot be easily mapped onto one other. This problem was addressed by using custom export methods that allow extensive user control over object description. Translation was often found to require both data synthesis and design synthesis, and these problems were resolved in different ways.

Within the projects, I used three approaches to the problem of translation. Firstly, the development of interpretive frameworks within MS Excel™ managed the transfer of geometrical information between CAD and analytical software. Secondly, in cases where programs were limited in their ability to import or export design data, scripting was used to import and export that data in immediately useful formats or to an intermediary MS Excel™ spreadsheet. Scripting was also used to encode rules for transformation. Thirdly, in projects such as the Travellers and the Bendigo Canopy, modeling techniques were used which ensured that geometry was in a format immediately useable within other deployments. Considerable knowledge about the work processes, requirements and design intentions of the architect and engineer underlaid the success of each approach, clearly illustrating the general point made in Chapter Five that information is only precise in the context of a deployment. This importance of precisely tuned information is further discussed in the following section.

In examining the second claim made in Chapter Five, that ‘high resolution’ analytical tools can support ‘low resolution’ methods for generative, performance-based design exploration, I have investigated interfacing CAD and analytical software to support generative performance-based design processes that develop interdependencies between architectural and engineering concerns. This investigation has involved either interfacing CAD with analytical software (structural and environmental), or using CATIA™, a program which incorporates both structural and architectural functions, to facilitate negotiation and co-authored design solutions. I have found that performance-based design can facilitate joint and creative architect-engineer problem solving, and that ‘high resolution’ analytical tools are compatible with exploratory approaches suitable to the early design phase.

Evidence for this compatibility is present in the Bendigo Canopy (2006), Venice Bridge (2007) and Cheese Tower projects (2007), which successfully closed the loop between proactive design and proactive analysis. A crucial issue found when testing this claim in the Venice Bridge and Cheese Tower projects, which used analytical software generatively, was that a generative approach
impacted significantly on the process of information exchange. The data exchanged between CAD and analytical software had to facilitate transformation, which required a high interdependence required between data transfer and data transformation. In order to close the design loop, the makeup and formulation of design data transferred between CAD and analytical software was informed by wider approaches to analysis and reinterpretation. This finding tends to support the claim by Shea that “the exchange of information between tools requires more than just transfer of geometric data” (Shea et al 2003: 555), and that closing the analysis-synthesis loop involves more than just the “juxtaposition of representations” (Koutamanis 2000: n.p.).

These projects revealed that two limits to interdependency which I identified can be resolved using 3D digital tools and techniques. The first of these limits was the time taken to construct a model for simulation. The second was the lack of integration between the analytical results and other aspects of the design process. However, solutions to these problems were reliant upon a third identified concern, that of the non-specialist’s ability to interpret correctly the information returned from analysis and to use it effectively in guiding design development. These projects relied on innovative approaches to simulation and analysis taken by Arup engineers, and on an extensive period of discussion and communication between the design team in which design goals were related to methods for achieving them. These facts draw us back to a question that I posed in the Introduction Chapter, being by what mechanisms should architectural designers engage with ‘performative logics’ (drawn either from analysis or fabrication) which, at least in an objective, measurable form, have not been part of the early design phase and arguably the modern architectural domain? The results suggest that the approach I have taken, working across the two domains, has been an appropriate one.

Thirdly, I claimed within Chapter Five that parametric design and scripting can facilitate a back-propagation of downstream information into early architectural and engineering design exploration. I have discovered that parametric design tools and scripting can embed at least two levels of useful information, heuristics and more detailed algorithmic types of information. I have also found that this approach supports at least three aspects of early design exploration: firstly, it responds to the problem of overly slow and accurate engineering feedback; secondly, it coincides with the engineer’s focus on the design of structural systems rather than of member sizes; and thirdly it facilitates the iterative exploration of design alternatives. Back-propagating design information across disciplinary boundaries enhanced the abilities of other designers (architects, engineers and fabricators) to solve common problems, which I have noted is a characteristic of interdependent processes.

In the Bendigo Canopy project, a structural sensibility was back-propagated into the geometry creation process through the encoding of structural rules of thumb or heuristics. Previously un-encoded knowledge was made available through an approximate but shared representation - a ‘common model’ within Rhino™. Although not a mathematically rigorous approach, the use of scripting made it possible to foreground a synthesis of architectural and engineering ideas without the problem of establishing a synthesis between disparate data sources. It also facilitated a highly iterative design process, which was shown to be compatible with a subsequent, more rigorous
analytical approach.

In the Travellers project, I found that the requirements of fabrication provided a common ground upon which architects and engineers can develop interdependent working processes. The Travellers project demonstrated that 3D digital methods can mobilize the capacities of contemporary manufacturing effectively. Scripting was used to facilitate early costing and material purchase and to encode downstream fabrication requirements. These requirements provided the basis for a modelling strategy that informed the design and structural analysis of the sculptures. Incorporating these requirements into the early digital process then led to significant benefits when fabrication commenced, including a drawing-less documentation process.

The forth claim in Chapter Five, that *3D digital design tools can enable representations that facilitate trust, and therefore interdependency, by increasing understanding and reducing perceived complexity, risk and uncertainty*, is perhaps the hardest to locate unambiguous evidence for. The benefits of representations that increased understanding and reduced perceived complexity emerged explicitly within the Travellers project but no others. This claim was also tested within the Rectangular Pitch Stadium project, however this trial revealed perceived risk as also dependant on perceived cost and other factors.

Within the Travellers project, 3D digital processes provided ways of automating the extraction of fabrication information directly from the 3D model, which provided unambiguous information upon which accurate costings could be made. The pre-purchase of steel that occurred within this project was not typical behaviour within the industry. Additionally, the process of encoding and communicating that information within the broader design process centralised the fabricator, who is normally a peripheral figure. The use of shared 3D digital model involved the entire design and fabrication team in joint processes of learning. The fabricator provided input into the development of the digital processes, meaning that he could understand and have confidence in the results. Had the same project moved through a traditional representation and procurement cycle then both that confidence, and the collaboration engendered, would not have occurred and arguably the project would not have met the constraints of time or budget.
7.2 PRACTICE-BASED OUTCOMES

The research that I have undertaken within this thesis affirms that 3D digital tools can enable interdependencies between crucial aspects of architectural and engineering design exploration during the early design phase. From within the context of practice, I have demonstrated several different approaches for using 3D digital tools to deliver outcomes which could not have been achieved effectively using non-digital means. Whilst specific in nature, these projects can be seen as indicative of the opportunities for architects and engineers to engage in earlier modes of design interaction and potentially new forms of collaboration. In each case, these projects represent solutions that neither party could have achieved working alone.

My research has not been directed towards a single method or process, but rather has explored an approach to enabling interdependency. As I have described in Chapter Two, the stance I have taken is that this approach is the only feasible one within a three year live practice context. I have found that this variable nature of the architect-engineer interaction is not a limit to enabling interdependencies between crucial aspects of architectural and engineering design exploration during the early design phase. While such an approach has prevented an in-depth exploration of any particular area, it has provided clear evidence that interdependency is not restricted to any particular problem or sphere of interest. Further, given that techniques migrated across projects, the projects provide evidence that an ad-hoc approach to the architect engineer relationship certainly does not mean ‘one-off ’ outcomes.

The research methodology I adopted has proven effective in facilitating my investigation. Firstly, working within the office of Arup Melbourne has enabled me to develop and test the research presented within this thesis whilst grounded within, and reflective upon, live practice. This unique perspective represents a real contribution offered by this thesis. Secondly, the decision to step outside traditional design literature and examine organisation theory has provided a productive way of initialising my concept of interdependency. Extending this framework through the experiences and understandings of practitioners and selected design literature has usefully connected the general themes developed back to practice-based limits upon, and solutions for, interdependency.

Through my research, the process of moving digital information between architectural and engineering software has been identified as complicated, particularly within the generative phase. We can ask legitimately whether this problem is simply a passing one. Already some identified problems of interoperability between CAD and analytical software are being solved by object-oriented approaches. Increased computational power may soon make other issues (specifically those associated with simplifying geometry) redundant.

However, a significant finding of this research is that the boundaries between architectural and engineering design practice are not set by software programs, but are negotiated and renegotiated with each project. The effective transfer of design information between different deployments
has, in each of the project cases, been informed by the problem at hand in addition to 3D digital tools and solutions on offer, and my research has found that there is a crucial relationship between ‘what’ and ‘what for’. This close relationship suggests that emerging object-oriented modeling approaches, which will certainly aid in integrating the work of architects and engineers, need to ensure that the ability to fully represent a building is not incompatible with more exploratory processes undertaken within early design exploration. Architects and engineers may well interact most effectively over common problems, rather than common objects.

The comments of Mr. A and Mr. C suggested that many of the obstacles to interaction occur because of a lack of knowledge about the work processes and requirements of the other involved parties. My findings tend to support this claim. At the core of each practice-based project has been the idea of matching differences across boundaries, and technology can only go so far (as evidenced in the Rectangular Pitch Stadium project). The success of these projects has relied on an understanding of requirements, intentions and methods of design that crossed domain boundaries. In each case, this understanding was developed through verbal communication, later supported by 3D digital tools. The conclusion that can be drawn is that attaining full benefit of the software is dependent upon parties communicating their requirements in advance, and that this benefit is dependent on having a sufficient awareness of those requirements to communicate them clearly.

In reflecting on the implications for practice, two seemingly contradictory conclusions can be drawn. The approaches to interdependency presented here have provided mechanisms by which engineers can ‘get into the process’ at a time when they have traditionally not been involved. Throughout the projects, 3D digital tools have involved architects and engineers in, at most, positions of co-authorship and at least increased involvement in the early design process. For some projects this interplay of architecture and engineering has resulted in more effective design delivery and for others in more effective design processes. In one sense, however, very little change has been recorded. The roles played by architects and engineers have remained distinct. Architects have continued to “do” architecture and engineers have continued to “do” engineering, as defined by the cultures of their respective professions. This outcome reinforces my definition of interdependency as a process which neither wholly reflects any one discipline nor joins different disciplines. The interests of neither party have been impinged upon or damaged, and the outcomes of the project work are beyond those achievable through independent working. From this, we can conclude that enabling interdependencies in the early design explorations of architects and engineers is a productive extension to the work of each, as well as to the results of both.

The research that I have undertaken for this PhD might be extended in two ways:

Firstly, by looking into the possibility of developing a Bendigo Canopy-style generative script within software such as Ecotect™ which allows for simple environmental analysis. This would link a heuristic structural intelligence with lighting, thermal and acoustic analysis in a multi-criteria,
generative, performance-based approach. My research suggests that this approach might provide a ‘Day 1’ process which would help to close the gap, at an appropriate level of resolution, between the early design exploration of architects and engineers. The findings from Bendigo Canopy, Venice Bridge and Cheese Tower would directly aid in the undertaking of such research.

Secondly, the research might be replicated within the context of architectural practice. While the nine interviewees included three architects, the research that I have conducted might not capture significant factors only observable from the architectural domain. Similar investigation from the architectural domain would potentially offer new methods for establishing interdependency and certainly allow for contrast and comparison. The practical realisation of these projects demonstrates that there is benefit in intersecting aspects of architectural and engineering design exploration during the early design phase. This thesis seeks to communicate the results of my research, learning and observation.

In conducting this research I have been able to learn about interdependency, and the processes that I have described might assist in supporting future productive interactions between architects and engineers. Of the project work presented within this thesis, at the time of writing one project, The Travellers has been realised and another, Marina Bay Bridge, is beginning construction. The interdependency that I have advocated within this thesis, facilitated by digital tools – all commonly available in contemporary architectural and engineering practice – have contributed substantially and demonstrably to the productive outcomes of these projects.
Appendix

Approaches to Interdependency:
early design exploration across architectural
and engineering domains
Appendix

A1. INTERVIEWS
A2. SOMETHING RICH AND STRANGE
A3. PUBLISHED PAPERS
A4. SCRIPTS
A5. ENLARGED DIAGRAMS
A1 INTERVIEWS

The following four interview transcriptions are reproduced here as a representative sample of the nine documented interviews. The other five interviews are available upon request. No part of these interviews may be reproduced without the permissions of both the author and the interviewee.

A1.1 MR. A INTERVIEW TRANSCRIPT:

WHAT TOOLS DO YOU USE IN THE CONCEPTUALISATION AND DEVELOPMENT OF A DESIGN IDEA?

Typically hand sketches are the fastest and easiest way to communicate. You can do them in real time as you are having a discussion with somebody. People who can sketch reasonably well can convey an idea very powerfully very quickly. I think engineers are quite good at sketching, architects seem to be very good at describing what they are trying to achieve in my experience. Sometimes an architect will make a lot of preconceived ideas about what he's trying to achieve, development work before he talks to anyone, and then just trying to understand which are the main drivers within the decision making process. But primarily early doors its hand sketching followed up by some simple visualisation.

95% of the buildings we do will be rectilinear buildings with a lot of repetition, and there are typically a very limited number of structural solutions for that which have been proven over time and are economical. It’s when there is something out of the ordinary that your focus stays wider. There are a lot of preconceptions in engineering that you have to be wary of - whether they are appropriate, whether they’re not.

For me, getting one of the CAD guys to draft up what looks like a structural diagram isn’t a particularly effective way of communicating a thought. I tend to do volumetric and perspective sketches to try to convey exactly what the thing is. When I become more proficient at 3D sketching, then I’ll do it that way but it’s essentially the same thing. But it’s not an engineering drawing. Maybe we ought to rethink how we represent our information.

I think 3D certainly helps, the greatest aid is in understanding the complexity: for the kind of things that we do, understanding complexity is half the battle. There are a number of structures that I’ve worked on in which its very difficult to represent those in a 2D form, and even a simple 3D representation you get a much better appreciation of how the thing might go together, how you might rationalise it to make it easier to build, easier to form. So there are certainly benefits in that respect. But in terms of communication between architects and engineers, I think the world is moving on from that now, and ideas about procurement and manufacture are where
the real strides are being made, because that’s where the money is being saved. Most architects and most of the engineers I work with talk the same language, generally, they talk a different language to manufacturers and a different language to contractors, typically. That part of the dialogue normally works reasonably well - it’s the rest of it – how do you from an idea to a representation in whatever format to something that’s real. That’s where the cost savings can be, that’s where the reduction in procurement times can come to the fore, its understanding how the design becomes a reality, and how you can model the cost, the constructability of it, are where I see the difference being made over the next 2 or 3 years. A couple of the projects I have worked on have made a big impact on that. Definitely the model is there.

HOW AND WHEN DO YOU COMMUNICATE YOUR IDEAS/RESULTS TO OTHERS?

It occurs pre tender, and post tender. But generally by the time you’ve gone to tender you’ve lost any opportunity – I’d say the month before tender goes out is really the exploration time. If you’re working with an architect that you’ve never worked with before, it takes some time for the relationship to develop, and quite often the problem is that there are a number of constraints which are unknown, and they only become known as you get into the process. One of the main ones is of course cost. Far too often the design freedom is there, you get to a point in time where a quantity surveyor will put a dollar value to it, and all of sudden its way too expensive. And this often happens just a few weeks before tender. And so you’re faced with the fact that the project is too expensive. That cost constraint isn’t there early enough, or it’s not understood early enough. You can’t blame the QS because he doesn’t know what you’re thinking about.

Things come in too expensive and people are forced to re-evaluate or value engineer. But then the pressure really is on because you’ve spent your fee, so you haven’t got the resources to commit too much to the process: but the scope for innovation is taken away quite considerably. Its understanding the project constraints and the desires of each party quite early. When you get that it’s fantastic.

WHERE ARE PROBLEMS OF TRANSLATION LOCATED WITHIN YOUR WORKFLOW?

A lot of this comes down to procurement routes – how these projects are procured. Typically, a design is produced and tendered without the input from the people who are going to manufacture and erect it. There is definitely a hole there, in the process, and we’ve talked at length in many jobs about “ok, what’s the format in which we’ll produce our data?” It really depends on how the contract is going to be let – if it gets let to tenderer A, you know the way their process works so you can tune your design to be in accordance with that. A very simple example of that is on some of the projects I’ve worked on in the UK we were pretty sure who was going to get the project, and for something simple like the design of wall steel beams into a column there are two principle ways you can do it: either thin plates or end plates, both perfectly
legitimate ways of doing it, both very economical. But one fabricator will have his machinery set up to do one, and another to do other. And you can bet your bottom dollar that if you design one way the guy who wins the contract will have his machinery set up to do the other. But if you knew in advance what that was going to be, you could tune your design specifically for that. And that’s part of the problem, you never actually know who’s going to get the contract at the end of the day. In my last large project, the whole premise there was to get the entire supply chain on board right from the outset, so design could be done, taking advantage of all of that input at an early stage. And I think it worked partially, but that is almost certainly the way forward. If you can build up the trust of all parties involved to go done that route, and particularly from the clients point of view that he’s going to get an economical thing at the end of the process without having to competitively tender it.

WHERE DO YOU SEE INTEGRATION OCCURRING OR FAILING TO OCCUR?

My thoughts since being here is that Australia is a few years behind the UK in quite a number of fields, things like the Egan report are now being successfully implemented in the UK now, but it means throwing the procurement process up in the air and rethinking the whole thing, and trying to bring the expertise of all parties on board, and informing the design process. There are benefits for engineers and certainly for architects to really understand how things go together, particularly complex forms, and how the initial design assumptions can be made to work far more efficiently.

Another example is two stadia in the US, both myself and the architect put an awful lot of work into the geometry of the stadium bowl, which was precast concrete, and again, tender process, tender return, and the pre-caster came back and said “it’s a real shame that you didn’t make those 4 things exactly the same by tweaking the radius by a few degrees, and instead of 14 different things you could have had 4”. Great advice, but it came too late in the day, and far too late to change the design because the program was so tight. And that’s probably another issue: as projects are being delivered far more quickly, the expectation is to deliver them quickly – there’s never time to explore these things once the contractors on board, its bang, bang, bang, deliver, deliver. So the expertise of these people, which is quite considerable, is often lost – it’s a fait accompli by the time they are on board.

The architect I worked with on a project just before coming to Melbourne was a co-located project, so we actually sat next to each other. And the response I got from him was that “I really didn’t understand what engineers did, because all we see from you is a very simple line diagram with bold lines and some less bold lines and some sizes of elements on them, and you wonder what on earth are we paying for, how difficult can it really be to produce those drawings. And it’s by having worked with you guys, and seeing what goes into that I understand that that’s just the end game, the final representation. But that doesn’t in any way represent the process to produce it, and the technical expertise that produces it.
A1.2 MR. B INTERVIEW TRANSCRIPTION:

WHAT TOOLS DO YOU USE IN THE CONCEPTUALISATION AND DEVELOPMENT OF AN IDEA?

We would use the same software to develop and test ideas as we would in the detailed design process, and the detailed analysis process, so that would be IES™ and the E+TA™ suite of software.

When it comes to testing ideas we would just go about modelling what we want to test in the same way that we would model a section of a building or a façade system.

An example would be the roof of an art gallery, where we looked at horizontal slats over a glass roof, we were looking at vertical louvre shading systems and testing the height and the depth of each slat versus the separating of each slat. To do that we used IES™, we built up a simple box with glass and these slats on top, and ran iterations for various thicknesses and separations, and looked at the amount of much solar energy was getting into the box, which made it a quite simple, straightforward relationship between the various components and how much energy was getting into the building. And that process, and all the various functions that we used in IES™, are exactly the same as if we modelled a building.

So I don’t think there is much of a distinction between software and techniques that we use to develop ideas and what we use to do ideas, because we are doing the same thing. At the conceptual level, we are doing a smaller scale with more iterations.

HOW AND WHEN DO YOU COMMUNICATE YOUR IDEAS/RESULTS TO OTHERS?

It generally starts with a workshop session, sitting around a table with a pen and paper, running through some ideas, and the architect might come up with an idea like “ok, I want a glass roof”, and then we would respond, saying “that’s ridiculous, its an art gallery, you’ve got to control the amount of solar energy that gets into that space”. So then we would batter around ideas of how we might incorporate different types of shading and glazing systems, and from that generally generate an exciting idea, like this idea of the slats over the glass roof, and then everyone would say “wow, that sounds amazing, how can we actually make that work”, at which point we would do a series of sketches that illustrate the idea, and then tell them that we will need to do some analysis on it, to test whether or not it is going to work.

At that point we would say give us a week or two weeks, go and build the model, run our iteration, develop a strategy as to how we will work out whether or not its going to be feasible, run the analysis, get the data together, analyse it and from that work out is it a goer or is it not. From that we would produce some sketches illustrating the solution of range of solutions,
because they are quick and easy to do, and use the software we can do some screen-grabs. At that point we would put together a feasibility study, otherwise just send through the images in a short email. And that would go to the architect and they would respond either positively or just not take it into account at all. Say we had 6 options, they might say ok lets just look at these two, and refine these further. At that point we would go back to the model, do deeper analysis or finer grain processing, and probably at that point do some serious checking to make that what we've modelled is scientifically ok, and then back and forth until we come to a solution that works. That process is always the case.

The tools are useful to produce images, to illustrate: if we are talking about geometry in design they are very useful because you can print out what it looks like, and they can understand it, but when it comes to things like how a space is actually affected by a design they're no good… [In this project] what we generated was graphs and hand sketches that illustrated how the space would perform in terms of humidity and temperature… The question the architects were asking was “how big do the holes need to be, and where should they be? We couldn't answer that with colour plots. Those are questions that can be answered in graph form.”

IVAM gallery there's an existing building there and they wanted to wrap a new façade around it, which was semi porous and create a semi outdoor condition between the existing building and the new façade - a huge perforated screen. We did studies on comfort, light etc, and tried to work out what the optimum properties for the porous membrane in terms of size of holes, which were considerably large, and material. Although we modelled it, we didn't generate any visual imagery for it. What we generated was graphs, which illustrated how the space would perform in terms of temperature humidity. We put these graphs together and that was our medium for delivering to our client, which was absolutely critical.

Along with the graphs, we used hand sketches to illustrate how the space would work: the questions that were asked were how big does a hole need to be, how many holes do we need to have – those are questions that can be answered through graph form.

WHERE ARE PROBLEMS OF TRANSLATION LOCATED WITHIN YOUR WORKFLOW?

Setting up the model, the geometry always takes a lot of time, and setting the parameters of how to use the model. Also determining the profiles takes a lot of time – there's a lot of stuff to click through. There's no automation, its all clicking forms. But also going back and checking, making sure you haven't clicked the wrong box.

If they do provide anything… for 3D geometry, if the architects done the 3D model, we would probably use that, but it's never a straight drag and drop. We might have to go around and rebuild all the surfaces. Contractors don't usually give us any information.
When it comes to façade system, E+TA has a really good library of materials, particularly for glazing, based on manufacturer's data. Similarly for IES, but not as extensive, Manufacturers do produce geometry, which is good for drawing up, but there's no intelligence embedded in there.

WHERE DO YOU SEE INTEGRATION OCCURRING OR FAILING TO OCCUR?

As designers, we are all working to achieve the same endpoint, and if everyone knows a bit more about how they do it, and what they do, the increased awareness of everyone's issues is beneficial to the end result: either in terms of you get there quicker, or you get there with better quality.

I think a good example is here – we all sit separately in our separate disciplines. So it's not a multidisciplinary team. In London, for example, the structural engineers sit with the electrical engineers, and in the short time that I've been here I see the difference: back in London people have a general awareness of what peoples issues are, whereas here there appears to be very little.

Physically we're separated, that doesn't encourage discussion, just seeing people work on their desk. Also, there's not a lot of multidisciplinary work that we get here, whereas in London it's been that way forever.
WHAT TOOLS DO YOU USE IN THE CONCEPTUALISATION AND DEVELOPMENT OF A DESIGN IDEA?

There are various levels at which we work: one is research, specifically developing and researching speculatively geometric systems, mathematical systems and other things that we find, from which we can generate forms that we can use in architecture or engineering. Just looking for opportunities. For example knots, topology, tiling, ways of creating modularity, folding systems, relational mechanisms, patterns and so on. We do work at every scale, from exhibitions, bridges, buildings, mostly collaborations,

Mapping is another level – mapping material systems onto forms, which is a problem that we commonly face particularly when working with other architects, for the serpentine pavilion we developed a number of algorithms to map a material system onto a form.

And then form-finding, which is creating form through methods of traditional structural formfinding, dynamic relaxation or methods of erosion, that come from structural efficiencies and structural behaviour. So in terms of structures, we are looking at interfacing geometric software with structural analysis software, optimising but also feeding back geometry.

I got involved in studying periodic tiling in the V&A project, where we looked at generation through subdivision – going from a single tile to 3 tiles, and keep subdividing. You can apply all kinds of rules to stop the algorithm when you reach a certain tiling density.

For the project with Aranda and Lasch, we started from a Danzer tiling system, isolated the points which are modular in both directions, and then created the Voronoi, which we scripted ourselves. I suppose we could have used QHULL™ but we needed the flexibility.

And then there are smooth forms. I developed this smoothing algorithm that uses Catenal-Clarke smoothing algorithms. Start with a crude typology, with a minimal number of facets, and then you smooth it. Also a smooth, streamline pattern. It's quite important when you use concrete. We are using this on a project with Ito.

But I think its important to distinguish the various levels – starting at the bottom, one is optimisation of the structure, one is mapping or geometric rationalisation, and one is the conceptual level where you go for something completely different, you don't start from the form. This is not literal, it is not form that is necessarily built, but it could be a programmatic diagram or a structural diagram. These are the first levels.
HOW AND WHEN DO YOU COMMUNICATE YOUR IDEAS/RESULTS TO OTHERS INVOLVED IN THE PROCESS?

Often what we do is to introduce a level of rigour. An architect might say that they want to do Voronoi, but you look at it and it's not actually Voronoi, or they might want to do a weave or smooth stuff but its never really that. And the issues of construction are never addressed, how do you take it to the fabricator, how do you describe it, how do you realise it. In those cases we introduce a notion of geometric rigour and knowledge of how things are built. Sometimes it quite hard – the architect might be very hooked on the image that they've sold to the client. It's one thing to win a competition, and another to make a project. Both are interesting, but it's not just a matter of making that image.

WHERE ARE PROBLEMS OF TRANSLATION LOCATED WITHIN YOUR WORKFLOW?

We suggested this technique right at the beginning – I had already tested it at Arnem transit hall, because it was the only way of creating its complex typology. I suggested it for this project as well because the way we formulated the solution at the competition level was by considering each of these catanoids individually, and then stitching them together, which was quite clumsy. What this does is to generate a smooth mesh, which addresses the problem of pattern as well as smoothness. I go from a crude mesh of quads to a smooth mesh of Nurbs. I use a technique for attaching data to a rhino object, where I record the crude mesh typology, which is made of quads, and then define a whole series of subdivision points which allow me to create smooth patches. It is a technique that ensures that the mapping or tiling is transferred from one to the other. You can attach all sorts of data.

We have written an interface with Sofistic™, which is the most powerful RC design software. At the moment Sofistic™ relies on a clumsy and unstable AutoCAD™ interface, which crashes all the time. Now we’ve managed to create any kind of surface and transfer it to Sofistic™ to analyse it. It’s going from crude rhino to smooth rhino with all the data attached. I’ve also developed interfaces to GSA™, so I don’t really use GSA™ anymore, instead I create what I call extended geometric models in rhino. It doesn’t need to be Rhino™, it could be anything. In a way Tristan Simmond’s .faf format is the most generic, which aims to capture all the data we ever use on top of the geometry.

When you convert geometry to a structural model its all the same – you have element properties and you have supports and loads, its kind of generic data. The thing is that we don't have to rely on any clumsy GUI in analysis software, we just have to deal with linking a powerful structural engine with a decent modelling tool, scriptable or whatever other way.
WHERE DO YOU SEE INTEGRATION OCCURRING OR FAILING TO OCCUR?

There is no typical way of working with the architect, there are only ad hoc ways that suit specific projects. We work with different architects who have different agendas and different expectations. We do engineering, but we do engineering plus. We give very good ideas about to construct the form, to model the form and analyse the form. In other cases we come up with entire geometric generators. If you work with an architect who is very formalistic then there’s almost no argument, they will start with an idea of what the building should be and then it’s a matter of making that building slightly smarter. Efficiency, the way you document it. Of more interest to us is when we can influence the concept, because then we go into something different that you moves away from that preconceived stylistic idea of what’s good or bad.

The points of intersection are completely grey area, and it’s up for grabs. Both architects and engineers are interested in that area. Ito’s interested in that from an architectural side, we’re interested that from an engineering side. It’s also a commercial position - basically there is a market there for new ideas, new geometric concepts, new principles that go far from the traditional engineer architect relationship. It’s a much bigger area, increasingly bigger. The tools are more powerful and the architectural horizon has folded back, we are generally looking at mathematics and nature not as a pure, formal investigation, but trying to find the intrinsic rules. It’s exploring new geometric systems, which can be organisational; geometric, spatial and structural.

We are a small group that sometimes relies on other groups for the delivery and management of large projects. We completely drive the projects though. Because we are small and we don’t have the exposure that we could have, clients might come to Arup to a high level director, who might then engage us. For other kinds of work we are known in the market and are a first point of contact. When we are selling services within that grey area it is easier go there and pitch our services because others will not be able to describe what we do. It needs to be a two way thing. It may be that there are small bits of larger projects of standard engineering projects that are suitable for our input. In the future, we will be the client’s point of contact, because we are effectively swimming upstream, because we are doing design, leaving aside the words architecture or engineering. There’s more interest in conceptual ideas than solutions. Part of our business case is to increase our exposure to the firm internally – we probably have a bigger external exposure than internally. We need to show the value we can deliver within the traditional ways and what we could gain from alternative modes of practice, which we are leading. When you have a bigger agenda you face bigger philosophical and corporate problems. If you look at Arup’s key speeches, he always talked about total design, and architecture was clearly always part of his journey, his remit, you just have to keep up with the times and see where those opportunities are in the modern world.
A1.4 MR. G INTERVIEW TRANSCRIPT

WHAT TOOLS DO YOU USE IN THE CONCEPTUALISATION AND DEVELOPMENT OF A DESIGN IDEA?

The thought of doing these kinds of projects without scripting is pretty horrendous. It's interesting how building types open up as possibilities once you start scripting. What you are seeing is a lot of people who are coming up with blobby shapes without understanding them, and to me they tend to be less elegant, less appealing.

I started off very, very small, and worked out how to do it and then used it. And kept using it. And then developed it and kept using it and then developed it and kept using it. I would have to say that by the time I started, around the age of 50 or so, by that stage it's quite difficult to pick up the programming skills that you really need to make this work. It takes a lot of effort and you are not going to get too many people doing it. I wonder whether they will do it with Generative Components™ (GC), I would hope so but I'm just not sure. In GC™ you have the option of working in a visual, CAD style way, and then to add parameters: a relatively painless process, doing it that way. To try to pick up Visual Basic™, which is relatively easy as a language, and to make it work will frighten most people off unless they're young and have been thinking that way for a while.

The challenge is that to get the most out of it you need the design experience and skills, and the programming experience and skills, and the vision, all brought together. That's quite a difficult combination to find. By definition you don't get the design experience and skills without doing it for some time. By the time you've done it you're probably past it in terms of doing the programming. Maybe the people who will make a success of this are those who started off doing the programming and then get the design skills and experience. Maybe we are half a generation away from it reaching a high point. How many older people do you meet that program and design and mix the two? Tristram is a prime example, but whom else? I don't know too many people who do that. Certainly I've not found people who do what I do. I can't think of any architect that I know who uses parametric tools from the inception of a design challenge. They may decide we are going to go this way and then develop the tools for it, but not those who develop a tool that you can use in a wide range of situations that is not too limiting. It's surprising really.

HOW AND WHEN DO YOU COMMUNICATE YOUR IDEAS/RESULTS TO OTHERS?

Everything that has passed through ArupSport in the 6 years that I have been here has been done parametrically from day 1. Being an architect is a slightly different situation from most people in Arup. It's not quite so much a case of serving the architect, and therefore you can behave in a rather different way. I can decide to investigate doing something like this, and I can
just do it. It gives one a degree of freedom that I don't think a lot of the engineers here are going to have.

I am not the best at collaborating in the sense that everyone in the world should use these tools. There are two sides to it: the first is that this is going to keep evolving, when you stop using it, it stops evolving and becomes dead and static. It will carry on for a while then die. I don't think that that's the best way to make use of these techniques. I think it's important that wherever possible things become live working environments. For instance you would have your working environment and be constantly evolving it, as I would mine. Where it is appropriate for them to interact that's what should happen. Our interaction at the moment is very basic: a 3d model that goes over and then CATIA™ is developed on the basis of that. In some ways it would be better to do them all in the one environment, but I don't think that is really practical, because it is horses for courses. The techniques that I started using, which were Microstation™ basic, excel visual basic and then Microstation™ visual basic once they updated it, have also been used by people like Fosters, who have done some of the most sophisticated stuff around. It is not really a friendly way of doing things, not really; however it is very good for some things. It has its place and CATIA™ has its place. Maybe we shouldn’t worry about it too much – it's too early to be heading for one environment. It's frustrating but perhaps that's just the way things are, and it will probably remain that way for 20 years. It's better than locking yourself into something like AutoCAD™, which people did far too early when it wasn’t very good and lo and behold they’ve stuck with it ever since.

WHERE ARE PROBLEMS OF TRANSLATION LOCATED WITHIN YOUR WORKFLOW?

If you take a line, start here and end here, there is a bit of parametrics here and a bit there, and in between there is a huge amount of manual and inputs plugging in all the way. It's efficient within that band and that bit’s not too bad, but if you want to change something here at the beginning you’ve got to go back, do all the manual bits again, and you have lost much of the benefit of doing it.

We are now within ArupSport talking about how we are going to integrate the various different environments that we have, which is really the stuff that I do, CATIA™ and GC, although GC is only just starting to get off the ground, because in my view it hasn't been viable in a serious way until recently. So we are talking about how we integrate those to create one overall process, joining up these huge gaps between parametric stages. With each project we move the bar up a bit, sometimes in reasonable chunks. Other times we are doing the structure always before we do the façade, but we are getting there. … That is much of the building, its not services yet, but it is dealing with all of the architecture, geometry and structure.

Who is going to look after the interface? GSA™ to Microstation™, Bentley is writing the
interface. That is one interface that will be a formal support. In terms of ArupSport we will do what we choose and what we can get round to... I tend to be horrified at the amount of time that we collectively spend doing things that are meaningless and wasted time. If it was programmed you’d never have to do anything other than hit the button and never have to do it again. The more of those we can get out of the process the more efficient and interesting we are going to make it. I am very keen that we start taking out more and more of the tedious and error prone parts of the process.

WHERE DO YOU SEE INTEGRATION OCCURRING OR FAILING TO OCCUR?

The earlier in the design process the more effective it tends to be. There is a huge market for Arup, for instance, to develop Day 1 tools. You have to build up those interactions, to find a way in to that process, because it is not normally led by the engineers anyway. It’s got to be something that the architects can work with - as an architect I would expect you to be developing tools that you use from day 1, solving design using them, and that let you collaborate with others. For me, it is very much the earlier the better.

I am keen to say we start right here. The other thing about starting right at the beginning is that it must be the most beneficial point in the thought process. The majority of the key decisions are taken there – do we put our effort into creating something at the end here, which way do we put a screw in the hole, or do we do it here. I haven’t yet seen enough people coming back to the point of conception; they are generally at a point beyond that.

Using precision tools is not a problem, provided you don’t expect to stay with those decisions – you just need to get started and then adjust from there. There are two sides to this: you want to be able to optimise very quickly and efficiently and know what you are working with, but you also want to free up the opportunities for investigating things that you would never normally dream of going near. To have the best of both worlds you want something very flexible that you can use up front, and also can do the precise bit as a by-product. The key thing is that it shouldn’t push you down one route, if it does then there is no real value. It should be as free as you can reasonably make it. That freedom might be that in programming it you have dealt with all the odd situations. A lot of people, when they program, they don’t worry about what happens when someone wants to divide by nought, that’s just one example. If you try to make it as reliable and robust as possible then perhaps you can go beyond 180 degrees, a point where previously it has fallen over, and it suddenly starts producing something you can use. It is that sort of thing that will open up the most possibilities.
A2. SOMETHING RICH AND STRANGE

Striking out from Flinders Street Station at an angle that almost obscures the presence of the Yarra River below, the Sandridge Rail Bridge suggests a design conceived at great remove from the site. Of course, the alignment was dictated by inflexible railway geometries, but after the rail line was closed in 1997 and the viaduct across Southbank demolished, the relic structure was an anomaly—a bridge that spanned but did not link, a massive structure carrying nothing, an axis delineated that points nowhere.

After years of abortive schemes for commercial redevelopments, in 2003 the Victorian Government and Melbourne City Council agreed to convert the bridge into an entire public space linking the redeveloped north bank next to Flinders Street Station to the new Queen’s Bridge and $15.5 million project complementing other preparations for the 2006 Commonwealth Games.

In 2005, Arts Minister Mary Delahunty announced a further $3 million for an artwork in this precinct commissioned from the Beirut-based architect Nadim Karam. The Travellers tells the story of migration to Victoria with 7.5 metre figures, each representing a ‘period’ and reflecting the bridge’s significance as a link to Australia’s first passenger railway line that carried thousands of migrants from Station Pier to Flinders Street station and to new lives in a new country.

Monumental historical and allegorical sculptures on bridges are not a new idea. It is impossible to forget Rome’s Pons S. Angelo or Prague’s Charles Bridge. However, Karam’s work makes this link directly. His 1996 Prague installation on the Mates Bridge created an obvious dialogue with the nearby Charles Bridge. In this, Karam reworked his theme of an ‘archaic procession’, a succession of timeless wanderers beyond claims of nationality, religion, race or territory.

The Travellers is another archaic procession, its migration program might be superficial re-bidding for local politicians, or ironic expression of the theme in a guise that depends upon nationality, religion, race and territory. However, Karam’s figures embody no weighty political gestures or contradictions. They are not archetypes glorifying the state, not historical figures that help define a local culture, not mythical archetypes that Joseph Campbell would recognize. Karam’s figures are global ‘urban toys’. Marmite-esque whimsy. Cute.

Cute and mobile. The Travellers move back and forth across the bridge in a sequence that [soon] be programmed to become a feature of daily life of Melbourne.

Again, the precedents are inescapable: the glockenspiel on Munich’s Rathaus, the automaton Moors on Venice’s Torre dell’Orarolo, any number of Black Forest cuckoo clocks.

If the word ‘kitch’ hasn’t come to mind yet, google Karam’s 10/1 Forest Species at Mile End Park in London (2001), a work akin to the Mecat Tudor Village and Fairies’ Tree Arts House.
in Melbourne’s Fitzroy Gardens. If the Forest Spirits can be explained away by Mike End Park’s emphasis on children and a compulsion to address children patronisingly in public places, consider his Carnival Elephant at Notting Hill Gate (2000), which looks like a rose wreath Keith and Kim might find for sale at Fountain Gate.

Victoria’s arts community must have reeled at their Minister’s announcement of one of the largest public art commissions in Melbourne’s history. Perhaps they took comfort in the knowledge that the divide between kitsch and art can be a fine line. Michelangelo’s Pisata - great art learned between madam cheesiness and questionable eroticism - proves the point.

Michelangelo - who, after a stormseason’s daughter, said he had taken in hammer and chisel along with her milk - also demonstrates the link between art and craft, something missing in much recent Melbourne public art. Localisation of The Travellers to drawings from Beirut reflects a separation now too common without such an excuse of geography. The disjunction between conception and execution means artists miss vital iterative relationships between ideas, medium and technique. Faced with intricate storytellers and earth-bound ballerinas, the public misses out on art’s expressive potential. In this environment a masterpiece in the traditional sense is impossible. Good art is a pig in a poke.

While craft-based art is passé, another tradition is held dear: the view that true art reflects an individual’s personal view of the world. So, as an artwork, The Travellers is attributed to Karam. Its fabrication is the labour of drones, accountable for administration and mechanics rather than a critical role in developing the concept.

Fortunately some drones don’t know their place. Rather than one artist’s personal creation, The Travellers is also the result of creative contributions by City of Melbourne architect Garry Ormston and many others.

The successful translation of Karam’s cartoon-like concept into three-dimensional form must be credited largely to Ormston. Working on the one hand with Karam’s Atelier Hapalus in Beirut - providing blueprints for the geometrical construction of drawings to make usable documentation for fabricators, increasing the size of the figures and steel sections to suit the scale of the site, working with aboriginal representatives to redesign their figures, and in many other ways - his collaboration reshaped the artwork. Working on the other hand with engineers at Acum and local steel fabricators, Ormston helped turn a formal concept based in found sketches into a structure that could only be built well using innovative software and mechanical technologies.

Karam’s earlier works use a silhouetted graphic style, grappling to represent ‘shadow-creatures’ with solid forms. In the Carnival Elephant he changed this to lattice work, an approach repeated in Melbourne, although the contrast between two contemporaneous works in the same medium by one artist could hardly be greater.

To me, The Travellers is still regrettably cute, but its construction has given the figures an acrobatic strength that makes them seem entirely at home on the superbly adapted Sandridge Bridge. Unlike the Ponte S Angelo’s angels, The Travellers’ robust form, gleaming steel and fine crafting suggest worldly associations from jungle gyms to Mies’s Barcelona chair. Akesi teapots to mobile dockyard creases. The theme of the archaic procession, the migration reminiscent - the whole merry conceptual cartload - has been transformed into a pearl.
A3. PUBLISHED PAPERS
Building design is a process often divorced from considerations about construction. Digital design methods are increasingly challenging the historic relationship between architecture and its means of production, but this extended reach is not necessarily accompanied by extended understanding or leverage of the production process. We present an urban sculptural project, The Travellers, in which digital techniques resolved critical issues of design, documentation and fabrication, but more importantly facilitated highly beneficial processes of negotiation. We suggest that this case based research has implications for future interactions between designers, makers and managers, shedding additional light onto issues of negotiation, responsibility, risk and trust that are often critical to the pragmatic undertaking of making.

Keywords: Design integration; digital design; fabrication; negotiation

Introduction

“The freer people are when they're thinking about a building, the more important it is to button down into how it's going to be made. Therefore we'll have to be working more and more closely with the people who are making it.”

Mike Cook

Complex geometries, limited budgets and short time scales place particular pressures on the design process, and successful outcomes are increasingly dependent on the capacity of the project team to synthesize, at an early stage, issues which occur across domains at different levels of precision. In examining the benefits that the computer brings to enabling design and its communication, we can ask to what extent the digital realm can support the confidence and buy-in necessary for working together more closely?

In this paper we suggest that the development and implementation of strategies for co-rationalisation whereby processes of design can intersect with processes of making, is a point at which the computer can discernibly aid in managing the more elusive issues of responsibility, risk and collaboration.

Project description

The Travellers are a series of 10 stainless steel sculptural figures designed by Nadim Karam and Atelier Hapsitus, constructed in 2006 on Sandridge Bridge, Melbourne. Each ‘traveller’ is a free standing stainless steel frame figure 7.5 m tall and between 5-10 m wide comprising of hundreds of connecting stainless steel RHS pieces, formed from ‘families’ of members that
make up two similar planar surfaces connected by a series of diagonal curves. Each planar member and its identical “twin” in the parallel plane (they average 750mm apart), are joined together by a diagonal member whose curvature in turn is defined by the two co planar members. The process of designing and constructing the Travellers was characterized by complex geometry, a limited budget and a project timescale of 9 months between conceptual sketch and being completed on site.

The project artist, Nadim Karam, was located in Beirut and the rest of the project team in Melbourne. This team included the Melbourne City Council as project architect and project manager, Arup structural and mechanical engineers and fabricator Silverstone Engineering.

In his recent review of the piece ‘Something Rich and Strange’, Ronald Jones discusses this divide between design and production, warning that “the disjunction between conception and execution means artists miss vital iterative relationships between idea, medium and technique”. While traditional notions of authorship fixate on conception and relegate production to “the labour of drones”, in this case the collaboration between drones is credited with the successful translation from concept to a well built “pearl” (Jones, 2006). The following account explores the issues involved and describes the collaboration that evolved during the project, revealing the process to be no place for drones.

Conceptual framework

Working collaboratively demands that “almost invariably we are working with others whose skills we lack or have only to a limited degree” (Thornton 2007, p. 102) however this relationship is limited by the fact that, with few exceptions, designers and makers tend to optimise within their own domains (CRC CI, 2002) and operate virtually discrete processes when designing the same building (Howrie 1995, p. 8). As Charles Eastman has noted, “the easy, close-working relationship between designers and builders has largely disappeared” (2004, p. 20), to be replaced by concerns over liability, responsibility and risk management.

It has been argued that the current design-bid-build method of contracting promotes litigation and restricts innovation (Eastman 2004, p. 21), and that the inadequacy of traditional drawing techniques has exacerbated the gap between “design and producing that opened up when designers began making drawings” (Mitchell and McCullough 1995). Kolarevic has added the view that “as digital data is increasingly passed directly from an architect to a fabricator, so will the building design and construction process become more efficient” (2001, p. 274) however simply substituting 3D models for drawings does not automatically guarantee a coherence of information, as Maher and Burry (2006, p. 202) have noted. Most of the cost and technical knowledge for the manufactured portion of buildings does not reside with designers but rather with specialty trade contractors and manufacturers.

The generation of trust, or buy in, by all parties is critically important to the success of any project, yet the amount of discussion concerning how this might be generated and supported lags far behind the default discussion of any BIM conference: how risk and liability might be covered (in the design world, claims to authorship generally do not correspond to acceptance of responsibility). Dennis Shelden (2002, p. 25), in discussing some of the reasons why Gehry Partners has pursued non-conventional methods to document and communicate design, explicitly addresses some of the barriers to buy-in:
“While fabricators could build the shapes, the process of bidding and coordinating the projects presented difficulty to construction managers. Accuracy of quantity takeoffs could not be guaranteed using conventional methods of measuring off the plans. Shop drawings – necessary for describing the detailed fabrication geometry – were difficult to render into orthogonal views... The limitations of understanding the project geometry through the lens of two dimensional views exacerbated perceptions of project complexity.”

Peter Rice has described addressing similar issues at Arup in the 70’s – “it is part of our procedure, in designing unusual buildings, to explain precisely what it is we’re doing... so that [all parties] understand that they aren’t taking exceptional risks” (1991, p. 104). These views suggest that at least part of the current concern about who holds responsibility for the accuracy of information is misplaced, and that instead one of the most immediate problems lies in developing ways to mitigate risk through increasing understanding. Processes and representational techniques that make it easier to understand and utilise design information are certainly central to addressing this problem. A second and rather obvious problem lies in generating accurate information in the first place.

Rationalisation, understood as the resolution of rules of constructability into project geometry (Shelden 2002, p. 78) comes in three forms (Fischer p. 13). Pre rationalisation defines the construction system before the design, whereas in a post rationalised approach the construction system is imposed after design has been finalised. Co-rationalisation sits between these two strategies, and occurs when the construction system is defined “alongside and to some extent through the process of defining a form” (Loukissas 2003, p. 32). Examples include the work of Gaudi (Burry 2003), Gehry Partners (Shelden 2002) and the Shoal Fly By project (Maher, Woods and Burry 2003). Co rationalisation requires that the designer and the fabricator work closely together during the design phase, with the result that making, and knowledge about making, informs both the design process and the generation and communication of design information. Such an approach has significant demands and impacts on the way we collaborate, some of which were encountered on the Travellers project.

**Project process / design**

To successfully respond to the issues of geometric complexity, budget and timescale previously described, the Travellers project required a design and procurement process that is unusual in an era of design-bid-build and competitive tendering. A key decision was made to include a specialist stainless steel fabricator as part of the project team from the beginning, so that practical fabrication advice was put into the process from the outset. This section examines how the impact of working ‘more closely with the makers’ affected the project, and describes the digital means by which this interaction was supported and its benefits incorporated into the design and documentation process.

**Early costing and material purchase**

The Travellers are made from stainless steel RHS, which is imported into Australia with a lead time of 12 or more weeks. This represented a third of the total project time, and it was quickly realised that it would be necessary to pre-purchase the steel before the design, analysis and fabrication processes had been finalised. Initial spline based 3D Rhino™ models had been supplied by the artist, which immediately provided rough schedules and costing information based on member lengths. After this information was extracted from the rhino models, the project manager (MCC) pre-purchased the stainless steel required to fabricate the sculptures - a quantity that was equal to all the stainless steel available in Australia at that time. As well as taking advantage of favourable pricing conditions, this removed a number of variables from the subsequent tender process; precise material lists (of which there was now a guaranteed supply) were supplied as part of the tender documents.
Full scale prototyping

The process of developing a modelling and representation strategy that best fit fabrication and analytical requirements also commenced at the project outset. Scaled wax rapid prototypes and 1:1 steel prototypes were generated from the 3D models to explore different connection options, which were discussed by the artist, architect, engineer and fabricator. The prototyping process gave all parties the ability to understand the requirements of the fabrication process and the geometric constraints that needed to be taken into account, and to deal with fine detail at a time when design had only just begun. As well as resolving issues about preferred connections and finishes, the prototyping process was instrumental in identifying where conflicts occurred between analytical and fabrication requirements, and in identifying strategies that minimised welding.

The largest and most complex sculpture was developed further than the others and provided the basis for a full scale prototype. At an early stage, it became obvious that it would not be possible to fabricate all the sculptures in the given time without additional geometric rationalisation of the radii used in the curving of the sections, particularly the larger 80X80 perimeter sections. This knowledge, as well as that gained about other geometric, representational and fabrication requirements, informed the design process for the other figures.

Geometric rationalisation

The artist’s rhino models used spline geometry which needed to be approximated with a standard series of arcs and straights for fabrication. Early prototyping had revealed that an additional level of geometric rationalisation was required. This second level of rationalisation was incorporated into the first, which had been coded as an automated process within Rhino™. The rationalisation process, scripted within the software, found the inflexion points along each spline, split the spline at these points and then matched each fragment with the closest arc from a standard series, the second level of rationalisation. Arches are separated by a straight line tangential to each – by inserting this minimum length line each member could be rolled as a continuous piece, greatly minimising time spent welding and polishing in the shop.

A second script generated each diagonal member. Geometrically, each sculpture is defined by two planes and a series of diagonal members that move between them, and these diagonal members can be generated directly from the corresponding members on the top and bottom planes. Automating the rationalisation process greatly decreased the time needed to model each sculpture, and resulted in a standardised set of parts that significantly reduced the time required for fabrication. Additionally, the process ensures accuracy and nodal connectivity between members, streamlining the analysis process.

Structural analysis

Because of the pre-purchase of steel, structural analysis for 9 of the 10 sculptures was performed after the stainless steel sections had already been acquired. 316 L stainless steel RHS was chosen as the primary material, and a sufficient degree of triangulation was incorporated at the nodes of steel member to member connections to enable for a relatively efficient structural system to be developed. Full strength butt welding at joints provide rigidity at all joints and also contributed significantly to the stiffness of the system.

Central to the structural design development process was the management of a single electronic file for each figure. This file was circulated in turn to the client, architect, and Arup designers, to be worked on/developed at each interface. Arup first rationalized the proposed geometry, and then analysed and checked these models for structural adequacy. Changes were proposed as necessary to ensure the sculptures were both buildable and structurally adequate, and these would be fed back to the client and ultimately the artist for their consideration. Convergence to a solution typically required two or three iterations to this process, and the City of
Melbourne played an important role as reviewer and in facilitating the speedy flow of information to and from the artist.

The Arup in house structural software GSA™ was used during the analysis, with input directly from Rhino™. The 3D model was prepared prior to importation into GSA™ so that all arcs were pre-faceted to control their approximation and to ensure that the geometric file was as “ready for analysis” as possible. This was important to model the geometry accurately and better estimate the structural performance of each traveller. The models were exported as DXF files; later design iterations and refinement generally involved tweaking, deleting, and inserting individual members, and not re-exporting the entire model.

Building without drawing
A series of protocols were developed to coordinate the modelling process between the various parties, which codified the ‘rules of engagement’. These included a specified information flow and required that information passing between the parties would be 3D models in specific formats. The digital models contained only centreline geometry and numeric annotations and, along with a series of spreadsheets, were the only documentation produced.

Documentation for each sculpture consisted of a 3D centreline model and an 8 page spreadsheet, one page per section type. A visual basic routine run from Excel™ extracted length and radius information from each member to the spreadsheet, and tagged each member and element numerically within the 3D model. The time taken to extract the spreadsheets was less than half an hour per sculpture, and the information they contained was fed directly into the bending machinery. The 3D model was used by the fabricators to check the numeric spreadsheet information on the shop floor. To aid the process of moving from spreadsheet to model, a scripted routine was written to select and zoom to any given member within the 3D model from the spreadsheet, as it was found that it was easier to locate members from the spreadsheet than to search for them within the 3D model.

Manufacturing process
While digital technologies were used extensively in getting 4455 stainless steel pieces onto the shop floor, the figures needed to be assembled, welded and polished by hand. While future projects seem certain to make use of emerging RFID tagging technology to help automate parts of this process, in this case a hybrid strategy of digital data and templating was employed. The digital extraction process numbered each piece, and this mechanism, in conjunction with plotting a full scale centreline template for each figure, assisted the fabricators to locate the members in each sculpture.

Close and early involvement from the fabricator, and the production of a full scale prototype, had a significant impact on the manufacturing process. The manufacturer of the prototype effectively ran workshops for the other three fabricators involved, going through the process and lessons learnt with each as the prototype was being built. This gave all fabricators, as well as the extended design team, confidence that the project could be systematically approached despite its apparent complexity, and again represents the highly unusual level of cooperation and information sharing achieved in this project.

Conclusions and implications for the future
Implicit and significant to this process was the level of engagement by the fabricator early in the piece, and the spirit of cooperation displayed by all disciplines to working towards favourable outcomes for the client and ultimately for all involved. A comparison with the traditional delivery process of jobs within the building industry reveals the following:

- Very few RFI’s were generated during the fabrication process, as design and buildability issues were generally resolved during the design process with the contribution of the fabricator. Significantly more information was available at the tender stage than is usual, reducing the perception of complexity and associated costs.
As information describing the full set out of each Traveller was passed on to the fabricator in an electronic format, shop drawings were not required at all. This significantly streamlined the fabrication process, as there was no requirement for the information to be reinterpreted, "handled" and checked a second time prior to enabling for fabrication. The absence of shop drawings meant that real savings were made in time, money and resources - while an estimated $20-30K was saved in fees for the production, review and finalisation of shop drawings, the real savings occurred in the avoidance of delay and the associated risk of liquidated damages.

The culture of cooperation consequently allowed for the efficient production and supply of information from the design team to the fabricators. This meant that the often confrontational and sometimes discordant nature of builder / fabricator /design team interplay observed on some building projects was non-existent in the delivery of the Travellers project.

In addressing the benefit that digital tools brought to the process of realizing the Travellers, we can immediately move beyond the low hanging fruit of increased efficiency and the 'file to factory' communication of complex forms. Rather, the point is that the entire design and fabrication team were involved in processes of learning, and if the same project had moved through a traditional representation and procurement cycle then both that, and the collaboration engendered, would not have occurred. In recognising that a conventional approach to procurement means digital tools cannot be used to their full advantage by the design and construction team, we raise a significant question: if the tools are to be used to their full advantage, then does it demand a non-traditional procurement process? In the opinion of the authors the answer is yes, because taking full advantage requires an education process involving the entire design and construction team which, in demanding increased involvement, transgresses many traditional boundaries. The observed benefits of collaboration make it difficult to return to the old ways.

Acknowledgements

The Travellers project was the result of the combined efforts of Nadim Karim and Atelier Hapsitus, Melbourne City Council, Arup, Silverstone Engineering, Able Engineering, DanFab, JW Metal Fabrications and Infracon. The research has been supported by the Australian Research Council, SIAL and the Embedded Research within Architectural Practice Linkage Grant, and Arup.

References


Loukissas, Y. Rulebuilding. Exploring Design Worlds through End User Programming. Master of Science
Abstract. This paper presents research on the relationship between digital tools and design communication, focusing on the interaction between architectural and lighting design. Early design integration often involves negotiating between different levels of resolution to inform a design that is still in formation, and part of the challenge is doing so in a manner appropriate to that phase of exploration. This paper describes some of the technical and social issues of translation and reports a project in which a generative design process supported the interaction between architectural design and lighting analysis; domains in which geometry is not necessarily a common ground.

Introduction

Traditionally, designers have used the same methods to design as they have to communicate that design; plan, section and model being the most common. Increasingly, this is no longer the case – design techniques, understood as the tools we use and the methods by which we use them, have progressed rapidly while the methods for design communication, though now digital, have remained stagnant. At the same time, the purpose of design communication is shifting, from that of dissemination to that of integration. Increasingly, the desire is to integrate external knowledge into the design process so as to inform it at an early stage.

The translation of information between design and analytic domains can be made more or less difficult by the representations used to communicate that information. These representations form the interface by which both parties interact. In the case of architectural design and lighting analysis, the mapping process has been more difficult, requiring significant rework, ‘cleaning’ and interpretation on both sides. Practically, the result of this is that analytic tools are typically used late in the process to confirm or deny a particular solution, while intuition and precedent guide early design iteration. Drawing has not provided a particularly good interface – indeed, Peter Rice has argued that many environmental loads, including light, cannot be adequately addressed by drawing, and attempts to understand them through drawing are likely to be fundamentally flawed (Robbins, 1994).
This paper reports the use of a generative design process to intersect ‘high resolution’ lighting analysis with ‘low resolution’ design exploration. The process described determines the optimum placement and sizing of window openings in a façade for a program configuration that is still open to manipulation. Working from the premise that, to a large extent, integration is a function of the efficacy of the mapping technique employed [mapping being the term used within practice to describe the translation and interpretation of information between programs and between domains] it presents evidence that mapping can beneficially be used as a constructive act that extends beyond the replication of geometry in semantically different domains.

Mapping: relating design and analysis

An early claim for the introduction of digital tools to the design process was that their increased precision would solve the problem of description (Mitchell, 1999); greater exactitude would eliminate miscommunication and result in an integrated design process. But this has not occurred, and the complexity of integrating design and analysis, specifically the role of interpretation within this process, has been the subject of ongoing research and literature. Several key issues have been identified:

A. That standard mappings do not support design simulation communication, which are typically highly project specific and require ‘expert translation’ (Augenbroe, 2001),

B. Interpretation rather than precision has limited design integration (Luebkeman, 1992), and that

C. Successful integration may require transformation: design and analysis frameworks may both need to be active for successful integration, rather than the typical situation where one is active and the other reactive (Johnson, 2004).

The focus on interpretation at the interface is twofold. There is firstly a technical problem; that typically CAD and analytic programs are semantically far apart. This translation is not well supported by existing mappings and therefore geometry useful in one deployment is not immediately useful in the other. A second problem is that information returned from analysis also requires ‘expert translation’ and interpretation to be developed within the design intent. For instance, analysis might reveal that a specific part of the façade receives a lot of sunlight throughout the year: should this mean larger windows or a greater extent of shading, a change in façade form or cladding or a change in the building’s programmatic layout?
Mapping: export as

Both engineers and architects are to an extent limited by the capacity of their tools for interpretation. CAD and analytic software typically affords a number of ways of saving a file, both through the various native file-types and exchange formats, through an ‘EXPORT AS’ type command. Generally the user chooses whichever file-type is native to the program that they will be exporting to, if available, or else chooses a certain file-type because it provides some particular affordance.

CAD programs do not have the explicit inverse of this command, ‘IMPORT AS’ – all importing is done through the all eggs in one basket ‘IMPORT’ command, where whatever information within the file that is semantically compatible with the program is extracted in a single way, and that which is not, ignored. Contrast this to opening a text file [.txt] with Excel, an exercise in constructing a mapping that may provide some insight into how an ‘IMPORT AS’ command in CAD might function. Via the import wizard, the user can determine exactly how the information within the text file should be mapped to the Excel spreadsheet: on what line the translation should begin, whether the relationship of text strings to cells should be determined by fixed widths or delimitation, how those delimiters should function, whether the cells should be formatted to reflect dates or times etc, etc. There are a surprisingly large number of possible string to cell mappings.

Mapping: in practice

The [self-described] workflow of an analyst using Radiance (Radiance is an industry standard ray-tracing tool used in lighting analysis) is shown in Figure 1.
It can be seen that while playing a significant part in the process, digital tools and techniques play little role at the interface between architectural and engineering domains. The transfer of information takes place primarily through verbal or graphic means, including false colour images (a technique for visualising datasets), tables and graphs, and sketches. This information is typically packaged as 2D images and graph within a report. The work of relating the design and analytic frameworks is carried out entirely by the engineer, and only a limited amount of intelligence is passed between the parties, making quick, iterative design difficult. For many types of analysis this has partly been a product of history, a result of the tools being so hard or ‘clunky’ to use.

The design problem

The design problem developed from a recently conducted interview, in which a description was given of the process undertaken to design a semi-porous skin covering an existing art gallery. The design intent was to create a semi outdoor space with its own microclimate, and one of the important design questions was “what were the optimum properties for the skin in terms of sizing, location and quantity of the holes?” In describing the role that digital tools played in the development and communication of the design information between architect and engineer, the interviewee, a mechanical engineer, stated that “The tools are useful to produce images, but when it comes to things like how a space is actually affected by a design they’re no good... The question the architects were asking was ‘how big do the wholes need to be, and where should they be?’ We couldn’t answer that with colour plots.” This project was particularly complex, given that many environmental factors were considered and simulated. However it would seem likely that there was enough information produced to have enabled a more direct relationship between analytic results and geometric
representation. But, as the following section details, this is not such a straightforward process.

**Low to high resolution**

The following sections detail the process of designing an optimised façade system, where the optimum location and sizing of openings and diffuser panels are generated directly from analytic information. The design proposal, for an inhabited bridge, contained a complex programmatic mix of gallery spaces, cafeterias, administration areas and workrooms etc, each of which required differing levels of natural light. Being in the early stages of design, where the configuration of these programs was not yet determined, led to a particular problem in the relationship required between design and analytic frameworks: different program configurations needed to be tested quickly without the need for reanalysis.

The CAD program involved was *Rhino* [with a later *Generative Components* alternative], and the analytic program was *Radiance*, an industry standard simulation tool for ray-tracing. The design process is a generative one, in which generative design is understood as the use of a structure, often in the form of computer code that utilises rules, variables, and external information to generate geometry. The designer designs the code, of which the geometry is an outcome, and by altering the inputs and relationships within the code can generate a large range of possible designs. In this case, the range of possible designs represents the total number of possible program configurations within the building, and their corresponding optimal window opening configurations.

**Cad to Radiance information transfer**

An initial unarticulated building envelope was designed within Rhino. This envelope was split into 750 X 750 mm polygons, each a possible site for a window opening. This grain was chosen because it was fine enough to stop the sunlight coming through any particular panel from flooding the interior, and it sat well within the range of standardised building components.

To accurately and efficiently reproduce polygons and the information associated with them in both CAD and analytic environments, it was necessary to retain a link between each façade polygon and the analytic results associated with it. With over 1300 possible positions for openings, it was essential to understand how to manage the large amount of information in an orderly way as it undertook two translations: from *Rhino* to *Radiance*, and from *Radiance* via *Excel* [where the analytic results were collated in a design file] to *Rhino* [or *Microstation* and *Generative Components*]. Tools that support this translation were found to be non-existent, as analytic information of this kind is used almost exclusively to produce visualisations either within the software or via programs like *ParaView*.

A rhinoscript was written that, for any given collection of n-sided meshes within *Rhino*, output 5 text files, each coordinated so that the information pertaining to any particular mesh always occurred on the same
The benefits of writing an export script rather than using a pre-existing file-type were:

A. Elimination of cleaning. The term ‘cleaning’ refers to the task of manually rendering a 2D or 3D CAD file importable for analysis. It typically involves stripping out superfluous information, geometric rationalisation and simplification, re-layering etc. to create an idealised version of the information.

B. Pre-application of materiality. Layering within Rhino was used to assign simple material definitions to geometry. By using strings recognisable to Radiance this intelligence could be carried over into the analysis model.

C. Control over naming. Naming each geometric entity provided a means to synchronise geometric entities with the analytic information associated with those entities.

D. Knowledge made explicit. Radiance is an idiosyncratic program, in that it is driven with a high level of customisation. Working closely with the analyst led to an understanding of exactly what information was required, and how to communicate that most effectively.

Relating the Frameworks

This section discusses how the analysis and design frameworks were related through the method of simulation, the collation of the results and the way by which those results provided the input data for geometry generation. Work in matching these frameworks had already begun, as detailed above, and one effect of this was that the model for simulation was geometrically the same as the architectural model. While this avoided the issue of multiple geometrically dissimilar models, it is important to note that there is no actual advantage in having the same geometry and in many cases it would be beneficial not to.

Because the design intention was to explore a solution space of possible program configurations, rather than analyse a single specific case, the analytic framework was structured so as to analyse the building as a series of generically programmed zones. By establishing the relationship between each zone and the facade, programming could later be made specific by the designer who could explore different programmatic possibilities by changing which program occupied which zone. The designer could then generate the optimum facade openings for that combination without the need for re-analysing.
The floor-space was divided into nine programmatic zones, each approximately 100 m². These zones were filled with a 1X1m grid of points, which became the points tested in the Radiance simulation (fig 2). A 1X1m grid of points was seen as giving a sufficiently accurate result: the more point locators the finer grained the results are, and the longer analysis takes.

Within the Radiance simulation, each point casts rays towards the ‘sun’ [a collection of its combined positions over the course of a year]. This amounts to thousands of rays for any point. When a ray passes through a façade mesh, a hit is registered for that mesh (fig 3). The more hits registered by a mesh the more direct sunlight reaches the point being analyzed via that façade element. Through this method the relationship, in terms of direct sunlight, between any particular façade element and any particular location within the building can be determined. Any single façade element typically affects multiple zones.

The results of this analysis were collated in a design file within Excel (fig 4). The design file is a database cataloguing each façade mesh and the number of hits recorded by it. Within the design file, the designer can test different programmatic configurations, in this case 6 programs types between the 9 zones. This combinatorial testing is achieved by the application of a multiplier specific to each program, which relates the amount of daylight reaching that program to its required daylight factor as specified by standards.

These calculations are used to determine the sizing of façade openings and diffusers. The results are filtered through 8 possible window sizes, which range from 100 to 800mm in width and step in 50mm increments. If the level of sunlight is calculated as being too sensitive for a
particular program, or of no benefit, no opening occurs. Behind the spreadsheet sits a *Visual Basic* routine that takes the calculated window width as an input for the geometry generation in *Rhino*. The script generates on average 1000 correctly sized and located window openings for any tested combination. In the *Generative Components* version, it also generates the diffusers, taking advantage of GC’s capability for mass instantiation of parametrically defined objects.

![Figure 5. Two configurations from the design file.](image1)

![Figure 6. Façade openings instantiated on architectural model.](image2)

**Conclusion**

Seeking to negotiate different levels of resolution is characteristic of the design process, but difficult to achieve. The project reported in this paper has described how digital strategies of practice can be used to construct such a mapping. It has presented some of the reasons why this extends beyond the direct translation of geometry in semantically different domains, the most significant being that analysis often involves modelling the problem rather than the building. This is particularly the case in seeking to integrate architectural and lighting design, where the issues involved are often not geometric.

The generative process described had the benefit of both quickly informing a design in a manner appropriate to early design exploration and of extending the inter-domain understanding of the parties involved. In comparison to more common ‘over the top’ modes of communication, such a process can avoid later re-engineering by the earlier integration of analysis, which assumes an active rather than reactive position within the design discussion.
Acknowledgements

The author gratefully acknowledges the contributions of Tai Hollingsbee and Jon Morgan of Arup Melbourne

References

Augenbroe, G. Building simulation trends going into the new millennium, in Seventh International IBPSA Conference, Brazil, 2001
Luebkeman, C.:1992, Good Fences Make Good Neighbors?, Architronic, 1(1.07)
A4. SCRIPTS
A4.1 RHINOSCRIPT (FOR BENDIGO CANOPY)

```
._RunScript ( Option Explicit

Sub DrawVoronoiDiagram()

Dim ptCloud, BBox, canopy, myDiv, linecollect, myEnd, EndY, myvar2
Dim arrPt, i, j, arrCrv, oncrv, myCrv, myVert, u, myDist, mybigdist, mybigdistnum, mysel, g, myvar, myLower
Dim myVertLower, strCmd, dblScalingFactor, MidX, MidY, MidZ, EndX, dblRotateFactor, startpt, Endpt, myAngle, angle1, angle2

' AUTOPILLOT
myvar = 2
myvar2 = 0.5

' WITH OPTIONS
'myvar = Rhino.IntegerBox ("Select roof depth factor", 2, "roof depth factor")
'myvar2 = Rhino.RealBox ("Select scaling", .5, "scaling factor")

'------------------user selections-----------------------------

ptCloud = Rhino.GetObject("Select a pointcloud...", 2, vbTrue, vbTrue)
If IsNull(ptCloud) Then Exit Sub
canopy = Rhino.GetObject("select the canopy surface")
BBox = Rhino.BoundingBox(canopy)
arrCrv = Rhino.ObjectsByType (4)
arrPt = Rhino.PointCloudPoints(ptCloud)

'------------------voronoi----------------------------
For i = 0 To UBound(arrPt)
    Rhino.EnableRedraw vbFalse
    VoronoiPolygon i, arrPt, BBox, 20
    Rhino.EnableRedraw vbTrue
    For j = 0 To UBound(arrCrv)
        oncrv = Rhino.IsPointOnCurve (arrCrv(j), arrPt(i))
        If oncrv = True Then
            myCrv = Rhino.SelectedObjects
            myVert = Rhino.PolylineVertices (myCrv(0))
            mybigdist = 0
            For u = 0 To UBound(myVert)
                myDist = Rhino.Distance (arrPt(i), myVert(u))
                If myDist>mybigdist Then
                    mybigdist = myDist
                    mybigdistnum = u
            End If
            mybigdist = mybigdist/myvar
            myDiv = Rhino.DivideCurveLength (arrCrv(j), mybigdist)
            ReDim linecollect (UBound(myVert))
            MidX = myDiv(1)(0)
            MidY = myDiv(1)(1)
            MidZ = myDiv(1)(2)
            EndX = myDiv(1)(0)+1
            EndY = myDiv(1)(1)+1
            startpt = Rhino.CurveStartPoint (arrCrv(j))
            Endpt = Rhino.CurveEndPoint (arrCrv(j))
```

```
angle1 = Array(startpt, Endpt)
angle2 = Array(startpt, Array(startpt(0), startpt(1), startpt(2)+1))

myangle = Rhino.Angle2 (angle1, angle2)

'work out what way the column is going

If startpt(0) = Endpt(0) Then
    If startpt(1) > Endpt(1) Then
        dblRotateFactor = 180 + myangle(0)
    Else:
        dblRotateFactor = 180 - myangle(0)
    End If
End If

If startpt(1) = endpt(1) Then
    If startpt(0) > Endpt(0) Then
        dblRotateFactor = 180 - myangle(0)
    Else:
        dblRotateFactor = 180 + myangle(0)
    End If
End If

dblScalingFactor = myvar2

myLower = Rhino.copyobject (myCrv(0), myDiv(0), myDiv(1))

strCmd = "_Scale 
strCmd = strCmd & MidX & "," & MidY & "," & MidZ & " 
strCmd = strCmd & dblScalingFactor

Rhino.UnselectAllObjects
Rhino.Selectobject (myLower)
Rhino.Command strCmd

'/////////rotating 3d

Rhino.UnselectAllObjects
Rhino.Selectobject (myLower)

strCmd = " _Rotate3d 
strCmd = strCmd & MidX & "," & MidY & "," & MidZ & " 
If startpt(0) = endpt(0) Then
    strCmd = strCmd & EndX & "," & MidY & "," & MidZ & " 
Else:
    strCmd = strCmd & MidX & "," & EndY & "," & MidZ & " 
End If

strCmd = strCmd & dblRotateFactor

Rhino.EnableRedraw vbFalse
Rhino.UnselectAllObjects
Rhino.Selectobject (myLower)
Rhino.Command strCmd

'////////////////

Rhino.UnselectAllObjects
Rhino.SelectObject (myLower)
Rhino.Command "-Planarsrf enter"
Rhino.UnselectAllObjects
myVertLower = Rhino.PolylineVertices (myLower)

'////////////////////
For u = 0 To UBound(myVert)
    linecollect(u) = Rhino.AddLine (myVertLower(u), myVert(u))
Next

For g = 0 To UBound(linecollect)-1
    Rhino.unselectallobjects
    mysel = Rhino.SelectObjects (Array(linecollect(g), linecollect(g+1)))
    If  mysel = 2 Then
        Rhino.Command "^Loft enter"
    End If
Next
    Rhino.EnableRedraw vbTrue
End If
Next

Next

Rhino.Print “Voronoi diagram complete”
End Sub

DrawVoronoiDiagram

Function VoronoiPolygon(index, datSet, BBox, gridCells)
    VoronoiPolygon = Null

    Dim midPt, arrPt, vecDir(1)
    Dim ptS(2), ptE(2)
    Dim ChordLength, Border
    Dim Grid(), vecS(2)
    Dim gridSize(1), gridOrigin(1)
    Dim brdLines()
    Dim i, j, x, y, N
    Dim p, q, g

    ChordLength = Rhino.Distance(BBox(0), BBox(2))
    ReDim Grid(gridCells - 1, gridCells - 1)
    For i = 0 To gridCells - 1
        For j = 0 To gridCells - 1
            Grid(i, j) = vbTrue
        Next
        Next
    gridOrigin(0) = BBox(0)(0)
    gridOrigin(1) = BBox(0)(1)
    gridSize(0) = (BBox(1)(0) - BBox(0)(0)) / gridCells
    gridSize(1) = (BBox(3)(1) - BBox(0)(1)) / gridCells

    arrPt = datSet(index)
    N = 0
    For i = 0 To UBound(datSet)
        If  i <> index Then
            x = Int((datSet(i)(0) - gridOrigin(0)) / (gridCells * gridSize(0)) * gridCells)
            y = Int((datSet(i)(1) - gridOrigin(1)) / (gridCells * gridSize(1)) * gridCells)
            vecS(0) = datSet(index)(0) + (datSet(i)(0) - datSet(index)(0))
            vecS(1) = datSet(index)(1) + (datSet(i)(1) - datSet(index)(1))
            vecS(2) = 0
            midPt = Array((datSet(i)(0) + datSet(index)(0)) / 2,
                          (datSet(i)(1) + datSet(index)(1)) / 2,
                          (datSet(i)(2) + datSet(index)(2)) / 2)
            brdLines = Array(midPt, vecS)
            grid(brdLines) = not grid(brdLines)
            N = N + 1
        End If
    Next
    Next

End Function
vecDir(0) = -(datSet(i)(1) - datSet(index)(1))
vecDir(1) = datSet(i)(0) - datSet(index)(0)
vecDir(0) = vecDir(0) / Rhino.Distance(datSet(i), datSet(index)) * ChordLength
vecDir(1) = vecDir(1) / Rhino.Distance(datSet(i), datSet(index)) * ChordLength
ptS(0) = midPt(0) + vecDir(0)
ptS(1) = midPt(1) + vecDir(1)
ptS(2) = 0
ptE(0) = midPt(0) - vecDir(0)
ptE(1) = midPt(1) - vecDir(1)
ptE(2) = 0
ReDim Preserve brdLines(N)
brdLines(N) = Rhino.AddLine(ptS, ptE)
N = N + 1
End If
Next

Border = Rhino.AddPolyline(Array(Array(BBox(0)(0) - 10, BBox(0)(1) - 10, 0), _
                                Array(BBox(1)(0) + 10, BBox(1)(1) - 10, 0), _
                                Array(BBox(2)(0) + 10, BBox(2)(1) + 10, 0), _
                                Array(BBox(3)(0) - 10, BBox(3)(1) + 10, 0), _
                                Array(BBox(0)(0) - 10, BBox(0)(1) - 10, 0)))

Rhino.UnselectAllObjects
Rhino.SelectObjects brdLines
Rhino.SelectObject Border
Rhino.Command "_CurveBoolean _DeleteInput=All _CombineRegions=No " & _
    Rhino.Pt2Str(Array(datSet(index)(0), datSet(index)(1), datSet(index)(2))) & _
    " _Enter", vbFalse

VoronoiPolygon = vbTrue
"Rhino.Command "planarsrf enter"
End Function

)
A4.2 SIMULATED ANNEALING SCRIPT FOR ECOTECT (FOR CHEESE TOWER)

randomseed(105513495577783414078330085995832946127396083370199442559)

iterations = 240
testBest = 0
StateBest = {}
MatBest = {}
matName = {}
objIndex = {}
objMat = {}

-- initialise basic calc settings
x = 6;
y = 6;
set("calc.precision", 4)
set("calc.sky overcast", 4500)
set("grid.max", 12000, 12000, 8500)
set("grid.size", x, y, 3)

--------------getting stuff---------------------
matName[1] = get("material.index", "Translucent_Skylight_01");-- get materials
matName[2] = get("material.index", "ConcreteRoof_Asphalt");

selected_objects = get("selection.count")-- get quads

for i = 1, selected_objects do
    objIndex[i] = get("selection.next", objIndex); --get object name
    objMat[i] = get("object.material", objIndex[i]); --get material index
end

--------------set up first configuration---------------------
for i = 1, 80 do
    l = random(1,selected_objects); --pick random quad
    set("object.material", objIndex[l], matName[1]); --change to open
    cmd("view.redraw");
end

--------------reset objMat array---------------------
for i = 1, selected_objects do
    objMat[i] = get("object.material", objIndex[i]); --get material index
end

------------------initial analysis-----------------------------
cmd("calc.lighting.grid daylight",true, 0)

initshading = get("grid.average")
print(“initshading is “..initshading)

------------------loop-----------------------------
for times = 1, iterations do
  --generate a neighbouring state

  concquad = {}
  openquad = {}
  concCounter = 1
  openCounter = 1

  --how many open, how many closed?
  for i = 1, selected_objects do
    if (objMat[i]==matName[2] ) then
      concquad[concCounter] = objIndex[i]--get num closed
      concCounter= concCounter +1
    else
      openquad[openCounter] = objIndex[i]--get num open
      openCounter = openCounter +1
    end
  end

  --change 1/2 num open to closed
  for i = 1, 40 do
    l = random(1, getn(openquad)); --pick random quad
    set("object.material", openquad[l], matName[2]); --change to closed
    ll = random(1, getn(concquad)); --pick random quad
    set("object.material", concquad[ll], matName[1]); --change to open
  end

  --reset name and material arrays
  for i = 1, selected_objects do
    objIndex[i] = get(“selection.next”, objIndex); --get object name
    objMat[i] = get("object.material", objIndex[i]); --get material index
  end

269
cmd("opengl.redraw")
print("there are closed "+getn(concquad))
print("there are open "+getn(openquad))

------------------analyse configuration-----------------------------

cmd("calc.lighting.grid daylight",true, 0)
avshading = get("grid.average") -- Compute its energy, (test it)

------------------compare-----------------------------

--comparison to find difference in energy

testVal = avshading/initshading

if (testVal > testBest) then --found a new best state, ACCEPT

for h = 1, selected_objects do
    MatBest[h] = objMat[h];
end

testBest = testVal;
print("recorded a new best at "+testBest);

else

dE = testVal
--T decreases from 5 to 1
if times > 0 then T = 5 end
if times > 40 then T = 4 end
if times > 80 then T = 3 end
if times > 120 then T = 2 end
if times > 160 then T = 1 end
if times > 200 then T = 0.5 end

testAccept = exp(-dE / T))

if (random() < testAccept) then
--NON BEST, but accept solution

for h = 1, selected_objects do
    objMat[h] = objMat[h];
end

print("non improving move "+testVal)


print("acceptance prob ".testAccept)
else
--go to statebest
for h = 1, selected_objects do
    objMat[h] = MatBest[h];
end
print("move to statebest at ".testBest)
end
end

cmd("view.redraw");
end

------------------and this was the best one-----------------------------

for i = 1, selected_objects do
    set("object.material", objIndex[i], MatBest[i]); --set material index to fully transparent
end

cmd("opengl.redraw")
Sub CATMain()
    Counter = 0
    For j = 1 To 24
        For i = 0 To 19 Step 4
            Dim documents1 As Documents
            Set documents1 = CATIA.Documents
            Dim partDocument1 As PartDocument
            Set partDocument1 = documents1.Item("bridge_bay.CATPart")
            Dim part1 As Part
            Set part1 = partDocument1.Part
            Dim hybridShapeFactory1 As HybridShapeFactory
            Set hybridShapeFactory1 = part1.HybridShapeFactory
            Dim hybridBodies1 As HybridBodies
            Set hybridBodies1 = part1.HybridBodies
            Dim hybridBody1 As HybridBody
            Set hybridBody1 = hybridBodies1.Item("Copy (" & j & ") of circ_setup")
            Dim hybridBodies2 As HybridBodies
            Set hybridBodies2 = hybridBody1.HybridBodies
            Dim hybridBody2 As HybridBody
            Set hybridBody2 = hybridBodies2.Item("Copy (" & j & ") of small_circ_top")
            Dim hybridShapes1 As HybridShapes
            Set hybridShapes1 = hybridBody2.HybridShapes
            Dim hybridShapePointOnCurve1 As HybridShapePointOnCurve
            Set hybridShapePointOnCurve1 = hybridShapes1.Item("Copy (" & j & ") of S." & 50 - i - Counter)
            Dim reference1 As Reference
            Set reference1 = part1.CreateReferenceFromObject(hybridShapePointOnCurve1)
            Dim hybridBody3 As HybridBody
            Set hybridBody3 = hybridBodies1.Item("Copy (" & j + 1 & ") of circ_setup")
            Dim hybridBodies3 As HybridBodies
            Set hybridBodies3 = hybridBody3.HybridBodies
            Dim hybridBody4 As HybridBody
            Set hybridBody4 = hybridBodies3.Item("Copy (" & j + 1 & ") of small_circ_top")
            Dim hybridShapes2 As HybridShapes
            Set hybridShapes2 = hybridBody4.HybridShapes
            Dim hybridShapePointOnCurve2 As HybridShapePointOnCurve
            Set hybridShapePointOnCurve2 = hybridShapes2.Item("Copy (" & j + 1 & ") of S." & 50 - i - Counter - 1)
            Dim reference2 As Reference
            Set reference2 = part1.CreateReferenceFromObject(hybridShapePointOnCurve2)
        Next i
        Next j
    Next j
End Sub

Dim hybridShapeLinePtPt1 As HybridShapeLinePtPt
Set hybridShapeLinePtPt1 = hybridShapeFactory1.AddNewLinePtPt(reference1, reference2)

Dim hybridBody5 As HybridBody
Set hybridBody5 = hybridBodies1.Item("Copy (3) of circ_setup")

hybridBody5.AppendHybridShape hybridShapeLinePtPt1

part1.InWorkObject = hybridShapeLinePtPt1

part1.Update

Next
Counter = Counter + 1

Next
End Sub
Figure 86. Structural model within CATIA, showing loads and restraints

Figure 87. False colour plot of stress analysis - optimised model
Figure 88. False colour plot of stress analysis - unoptimised model

Figure 96. Multi-criteria optimisation - false colour plot showing stress
Figure 97. Fixed branching points along the short sides of the canopy within the CATIA model
Bibliography


Aish, R; 2005, From Intuition to Precision, in Digital Design: The Quest for New Paradigms, Proceedings of the 23rd eCAADe Conference, Lisbon, pp. 10 - 14


Alexander, C; 1971, The State of the Art in Design Methods, DMG Newsletter, Vol. 5 No.3, pp. 3 - 7

Amor, R. & Faraj, I; 2001, Misconceptions about Integrated Project Databases, in ITcon, Vol. 6, pp. 57 - 68


Arup, O, 26.10.72 Institute of Civil Engineers


Baier, A; 1986, Trust and Antitrust, in Ethics, Vol. 96, No. 2, pp. 231 - 260

Balmond, C; 2002, Informal, Prestel, London

Balmond, C; 2004, An Aesthetics of Calculus: Round Table Discussion, in N. Leach, D. Turnbull and C. Williams (Eds.) Digital Tectonics, Wiley Academy, London


Barrow, L; 2001, Evolving Communication, online resource

Barrow, L; 2004, Elitism, IT and the modern architect opportunity or dilemma, in Automation in Construction 13, Elsevier, pp. 131 - 145


Bellinger, R; 1998, Professional development sessions dominate; new poll on engineers’
image released - PACE conference hits ‘a year of transition’, in Electronic Engineering Times,
September 14


Bennett, J. & Peace, S; 2006, Partnering in the Construction Industry: A code of practice for
strategic collaborative working, Butterworth-Heinemann

Online resource http://www.aia.org/aiarchitect/thisweek05/tw0930/tw0930bp_notjusttech.cfm
[accessed February 2008]

Billington, D; 1983, The Tower and the Bridge: The New Art of Structural Engineering, Basic
Books, New York

Billington, D; 2003, The Art of Structural Design: A Swiss Legacy, Princeton University Art
Museum, Princeton

Brady , M; 1999, Review: Interdisciplinarity; Essays From the Literature, in The Educational
Forum, Vol. 63, No. 3, Spring 1999

Brown, J, 2000, Design Plans, Working Drawings, National Styles: Engineering Practice in Great

Brazier, F. & Wijngaards, N; 2002, Role of Trust in Automated Distributed Design, in J. Gero
& F. Brazier (Eds.) Agents in Design 2002, Key Centre of Design Computing and Cognition,
University of Sydney, pp. 71 - 83

Bronfenbrenner, U; 1979, The Ecology of Human Development: Experiments by Nature and
Design, Harvard University Press, Cambridge MA

Burry, J, Felicetti, P, Tang, J, Burry, M. & Xie, M; 2004, Dynamical Structural Modeling: A
Collaborative Design Exploration, in Architecture in the Network Society, Proceedings of the
22nd eCAADe Conference, Copenhagen, pp. 312 - 317

Burry, J, Burrow, A. & Burry, M; 2005, Upholding the Poetic in Design Collaboration, in
CAADRIA 2005, Proceedings of the 10th International Conference on Computer Aided
Architectural Design Research in Asia, New Delhi, Vol. 1, pp. 288 - 299
Burry, M; 1997, Narrowing the Gap between CAAD and Computer Programming: A Re-Examination of the Relationship between Architects as Computer-Based Designers and Software Engineers, Authors of the CAAD Environment, in CAADRIA’97, Proceedings of the Second Conference on Computer Aided Architectural Design Research in Asia, Taiwan, pp. 491 - 498


Chaszar, A; 2006, Blurring the Lines: An Exploration of Current CAD/CAM Technology, in André Chaszar (Ed.) Blurring the Lines: Computer-Aided Design and Manufacturing in Contemporary Architecture, Wiley Academy, West Sussex, pp. 6 - 15


Emmitt, S. & Gorse, C; 2006, Communication in Construction Teams, Taylor and Francis, New York

Faber, C; 1963, Candela the Shell Builder, Reinhold, New York


Gero, J; 1998, Conceptual Designing as a Sequence of Situated Acts, in AI in Structural Engineering, pp. 165 - 177


Gross, M; 1986, Design as Exploring Constraints, PhD dissertation, MIT, Cambridge MA


Handy, C; 1976, Understanding Organisations, Penguin, London


Howrie, J; 1995, Building modelling: an architect's view, in Bepac Newsletter, Building Environmental Analysis Club, No. 12, pp. 8 - 10


Jones, J; 1977, How My Thoughts about Design Methods Have Changed During the Years, in Design Methods and Theories Vol. 11 No.1, pp. 48 – 62

Jones, R; Something Rich and Strange, in AR Australia, Vol. 96, pp. 28 - 30


Killian, A; 2006, Design Exploration through Bidirectional Modeling of Constraints, PhD dissertation, Massachusetts Institute of Technology, Cambridge MA

Klein, J; 1990, Interdisciplinarity: History, Theory, and Practice, Wayne State University Press, Detroit


Klinger, K; 2001, Making Digital Architecture: Historical, Formal, and Structural Implications of Computer Controlled Fabrication and Expressive Form, in Education & Curricula – 10 Digital Design Media, pp. 239 - 244

Kloft, H; 2006, Engineering Form, in A. Chaszar (Ed.) Blurring the Lines, Wiley Academy, West Sussex, pp. 82 - 96


Kvan, T; 1999, Designing Together Apart: Computer Supported Collaborative Design in Architecture, PhD dissertation, The Open University, Milton Keynes, England

Kvan, T; 1997, But is it collaboration?, in B. Martens, H. Linzer & A. Voigt (Eds.), Challenges of the Future: 15th ECAADE Conference, Vienna, p. 5, (CD ROM publication)


Larsen, O. & Tyas, A; 2003, Conceptual Structural Design: Bridging the Gap between Architects and Engineers, Thomas Telford Ltd

Lawson, B; 2004A, Oracles, Draughtsmen, and Agents: the nature of knowledge and creativity in design and the role of IT,


Leatherbarrow, D; 2005, Architecture's Unscripted Performance, in B. Kolarevic & A. Malkawi (Eds.) Performative Architecture: Beyond Instrumentality, Spon Press, New York, pp. 5 - 20


Luebkeman, C; 1992A, Good Fences Make Good Neighbors?, Architronic, 1(1.07) online resource http://darkwing.uoregon.edu/~struct/resources/essays/good_fences_make.html [accessed February 2008]

Luebkeman, C; 1992B, The Development of Structural Form, Architronic, 1(1.07) online resource http://darkwing.uoregon.edu/~struct/resources/essays/influences_on_choice_1.html [accessed February 2008]


MacDonald, A; 1997, Structural Design for Architecture, Architectural Press, Oxford


Marsh, A; 1997, Performance Analysis and Conceptual Design, PhD dissertation, University of Western Australia


McLeod, C; 2004, Seamlessness: Common Desire for Pure Structure, in L. Wallis, P Whitman & S. Savage (Eds.) TAKE 4 - Collaboration & Coalition: Creating Architectural Knowledge in Contemporary Practice, Royal Australian Institute of Architects, Canberra, pp. 25 - 32


Sasaki, M; 2005, Flux Structure, Nobuyuki Endo, Tokyo


Ortega y Gasset, J; 1932, Revolt of the Masses, G. Allen & Unwin, London


Peters, T; 1987, Transitions in engineering: Guillaume Henri Dufour and the early 19th century cable suspension bridges, Birkhauser Verlag, Boston


Peters, T; 1992, Bridging the Gap: Points of Contact Between the Architect and Engineer, in ARCHITRONIC, Vol. 1 No. 1.09, online resource http://architronic.saed.kent.edu/v1n1/v1n1.09.html [accessed February 2008]


Quality Assessment Report for University of Bath, Civil Engineering, March 1998, online resource http://www.qaa.ac.uk/reviews/reports/subjectLevel/q171_98_textonly.htm [accessed February 2008]

Rice, P; 1998, An Engineer Imagines, Artemis, London


Robbin, T; 1996, Engineering a new Architecture, Yale University Press, New Haven CT


Rowe, P; 1987, Design Thinking, MIT Press, Cambridge MA


Salvadori, M; 1959, The Engineer and the Artist, in Perspecta, Vol. 5, Yale School of Architecture, pp 16 - 19

Salvadori, M; 1991, Architects versus Engineers (Introductory chapter), in Bridging the Gap: Rethinking the Relationship of Architect and Engineer, New York, Van Nostrand Reihold, New York


Simon, H; 1973, The Structure of Ill-Structured Problems, Artificial Intelligence 4, pp. 181 - 201

Simondetti & Brodkin, 2005, 3D Documentation Transition: Draft 1, internal Arup report


Suchman, L; 1995, Making Work Visible, in Communications of the ACM (Association for Computing Machinery), Vol.38 No. 9, pp. 56 - 64


Thomas, G. & Thomas, M; 2005, Construction Partnering and Integrated Teamworking, Blackwell, Oxford


Williams, C; 2004, Design by Algorithm, in N. Leach, D. Turnbull & C. Williams (Eds.) Digital Tectonics, Wiley Academy, West Sussex, pp. 78 - 85

Xie, M, Huang, X, Tang, J. & Felicetti, P; 2005, Recent Advances In Evolutionary Structural Optimization, Keynote Lecture for Frontiers of Computational Science Symposium, Nagoya University, Japan