Contribution Structures
for
Requirements Traceability

Orlena Cara Zena Gotel

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in the Faculty of Engineering of the University of London. Also submitted for the Diploma of Imperial College of Science, Technology and Medicine.

August 1995
For David ... 
Your investment matures at last?
Abstract

This thesis is concerned with the problem of *requirements traceability* (RT). In the context of systems and software engineering, RT refers to the ability to describe and follow the life of a requirement in both a forwards and backwards direction. Despite being highlighted as an area in need of improvement, being introduced as a mandatory activity in development standards, and a recent surge in dedicated support, RT remains cited as a dominating problem by industry.

In recognising that RT problems are multifaceted in nature, and that there is unlikely to be an all-encompassing solution, we attribute their persistence to the lack of any thorough problem analysis. Prevailing approaches to RT are inadequate because, being primarily solution-driven, they fail to account for the underlying issues which perpetuate RT problems in practice.

Based upon findings from an empirical analysis of the RT problem, we suggest that the crux of the RT problem revolves around the paucity of information about the human sources of requirements and requirements-related information. We found that practitioners predominantly claim to have experienced RT problems when, being unable to retrieve information they want about requirements, they have further been unable to identify those in a position to supply it. We therefore propose that, by making the social structure that gave rise to requirements explicit and open to interrogation, a suitable basis would be provided for tackling this central issue.

To provide such a basis, this thesis presents an approach through which the *contribution structure* underlying requirements artifacts can be defined and maintained. This enables conventional forms of *artifact-based* RT to be extended with contribution structures to accommodate the diverse forms of *personnel-based* RT that practitioners were found to need. Experience with a prototype tool and a case study further demonstrates how the approach provides both a feasible and practical solution to the crux of the RT problem and, in so doing, corroborates the position of this thesis.
i. Statement of Contribution

We believe this thesis has made a contribution to the field of requirements engineering (RE) and, more generally, to the discipline of systems and software engineering. This contribution takes the form of providing an increased understanding of a notorious problem area, namely requirements traceability (RT), coupled with a practical approach for tackling what was found to be the central issue, namely the identification of those involved in the production and refinement of requirements artifacts. Although this contribution is described fully in the conclusions, we summarise the main components below:

(1) A review of the state-of-the-art in RT, including a critique of current support, plus a description of contemporary research.

(2) An empirically grounded analysis of what the so-called "RT problem" actually is.

(3) The differentiation of two fundamental types of RT, pre-RT and post-RT, along with an explanation as to why there is a need to focus more research on the issues related to pre-RT.

(4) Uncovering the shortcomings of conventional artifact-based RT and revealing the crux of the RT problem to be identifying the human sources of requirements and requirements-related information.

(5) The development of an approach to tackle the crux of the RT problem, based on modelling and using the contribution structure underlying requirements artifacts, which makes personnel-based RT possible.

(6) Demonstrating the wider impact of the approach with regard to issues like quality development and process improvement.
ii. Papers and Reports

A few technical reports have been written and papers have been published during the course of the author's Ph.D. studies. These are listed in chronological order below:


iii. Acknowledgements

This thesis would not have been written without the encouragement and guidance of my supervisor, Anthony Finkelstein. Thank you Anthony for picking me up out of the depths of despair, for giving me back my self-confidence, and for making sure I saw things through. It was wonderful to know there was always someone ready to listen to my ideas and help me shape them.

I would like to thank the other members of the Distributed Software Engineering Group at Imperial College, particularly Jeff Kramer and Jeff Magee, for providing a stimulating research environment. To Stephen Morris and Bashar Nuseibeh I owe extra special thanks for many discussions and comments on thesis drafts. To Keng Ng and Kevin Twidle I owe thanks for answering all the sorts of technical questions that arose from frequently working sixty miles away from college. I would also like to thank the members and affiliates of the Centre for Requirements and Foundations at Oxford University where my research began. Thanks especially to Marina Jirotka and Paul Luff, whom I hold entirely responsible for my sociological leanings.

Perhaps the group of people to whom I owe the most thanks are the practitioners I was fortunate enough to have access to. Firstly, thank you to all those who took part in my early data gathering exercises, and then to all those who were involved in the project used for my case study. Secondly, thank you to all the members of the RE group at Philips Research Laboratories for showing so much interest in my work, and for providing the opportunity to take it further. From this group, I would particularly like to thank Paul Gough for his amazing enthusiasm, as well as for his comments on thesis drafts. Finally, over the years I have had much contact with other practitioners, tool vendors, and academics. Each one of these people has influenced my work in some way or another. They know who they are and I thank them all for their contributions.

My family and friends have been brilliant! Moo - for always being ready to listen to my worries, cheering me up, and for believing in me. How I cherished those six o'clock phone calls. Dad - for pushing me, sometimes too hard, to get to where I am today. I also have you to thank for an eclectic set of text books. Mr X - well, for doing the kind of things that brothers are supposed to do, like driving me mad now and again with your visits. Patrick - for drawing me the grooviest posters. Nanta - for being so proud of me - how could I let you down? Paula - for hundreds of distracting emails. Gideon and Ali - for all the needed breaks in Hay. The list could go on ...

However, only David has had to put up with me every single day whilst I have been doing this thesis. I have never known someone to have so much tolerance - I was often unbearable - even I would not have put up with myself! When I moaned and groaned, he always managed to smile. Whenever my batteries ran low, he always found some novel way to recharge them, which I dare not go into! Thank you my love - you are the best friend I could ever have - and infinitely more!!!
iv. Financial Support

I would like to thank the States of Jersey Education Department, Channel Islands, for their continued financial support throughout my undergraduate and postgraduate days of study.

v. Trademarks

The rights to any registered trademarks used in this thesis are acknowledged.

vi. Disclaimer

The views and opinions expressed in this thesis are not necessarily those held by any of the above organisations or individuals. Any errors are, of course, entirely the responsibility of the author.

vii. Thesis Road Map

This thesis is divided into five main parts and is complemented by a number of appendices. Here, we outline the content of each of these parts, and so indicate the flow of the thesis argument. In reading this thesis, it is important to note that the abbreviation "RT" is frequently used in place of "requirements traceability", and that the abbreviation "RE" is frequently used in place of "requirements engineering".

Part I: Introduction

In Part I, we provide an introduction to the thesis. We first present the structure of the thesis argument and then include background material. This material locates the topic of RT within the disciplines of systems and software engineering and, in particular, within the field of RE. We discuss issues like why RT is needed, how it is currently achieved, and the direction of recent work in the area. The purpose of Part I is to highlight the RT problem, to motivate the need to tackle it, and to suggest the most appropriate way forward.

Part II: Problem

In Part II, we explore the underlying nature of the RT problem. We first describe both why and how an in-depth problem analysis was carried out. We then present the main findings of this work. Notably, we introduce a distinction between pre-RT and post-RT, to demonstrate why an all-encompassing solution to the problem is unlikely, and also to provide a framework through which to understand the issues involved. We explain how the majority of the problems currently attributed to RT are due to inadequate pre-RT and argue why there is a primary need for improvements here. We list the main barriers confronting such improvements and highlight ways in which some advances could be made. Drawing from this analysis, we locate what lies at the
heart of longer-term RT problems, namely the problem of tracing those individuals and groups who have been involved in requirements production and refinement. We then outline what is needed to address this focal aspect and mention a number of areas from which insight can be gleaned. The purpose of Part II is to clarify the nature of the RT problem, to expose where the most leverage for improvement lies, and to suggest how these improvements could best be realised.

**Part III: Approach**

In Part III, we present an approach to address the crux of the RT problem. We first describe the requirements for such an approach, paying particular attention to the limitations with current practice and the "political" issues likely to affect the acceptance of any proposed changes. We then introduce an approach based on modelling and using the *contribution structure* underlying requirements artifacts. We mention the assumptions it makes, explain the terminology it uses, and detail the steps it involves. We further clarify the approach using a model-based specification of its operation and go on to demonstrate how it provides the potential to answer the kinds of question frequently asked regarding those involved in RE. The purpose of Part III is to describe an approach to tackle the crux of the RT problem, to explain its basis, and to specify how it operates and addresses the problem in theory.

**Part IV: Validation**

In Part IV, we investigate whether the above approach addresses the crux of the RT problem in practice and, by so doing, whether it is likely to have an impact on alleviating RT problems. We first outline our evaluation method and assessment criteria. We then describe a prototype tool that was developed to demonstrate, refine, and test the approach. We show how this tool was itself designed in a principled manner, preceded by an RE exercise, and also sketch its final architecture. We follow this with a scenario, used to illustrate the operation and capabilities of the approach, and then with a case study making use of real industrial data. Based on these experiences, we attempt to evaluate the approach, its application, and its suggested implementation. The purpose of Part IV is to check whether our thesis is sound, to describe our method for doing this, and to provide a critical evaluation as a result of our findings.

**Part V: Conclusions**

In Part V, we set out the conclusions of the thesis. We first re-examine the thesis argument to determine whether or not the thesis position has, in our opinion, been successfully argued. We then discuss the contributions that have been made and the implications that these have for both research and practice. We point to the limitations, outstanding issues, and future research. Finally, we mention how the approach is currently in the process of being transferred to an
industrial setting. The purpose of Part V is to indicate how this thesis has made an impact in the area of RT, to show the next steps forward, and to end with a summary.

Appendices

The appendices are provided in a separate volume and contain material which supplements the main body of the thesis. This material ranges from a synopsis of our early data gathering exercises, through more examples relating to our modelling of contribution structures, to a summary of the data underlying our case study.
## Outline Table of Contents

### Part I: Introduction

1. Overview of Part I ................................................................. 34
2. Thesis Argument ................................................................. 34
3. Requirements Traceability ...................................................... 35
4. State-of-the-Art in Requirements Traceability.............................. 41
5. Critique of the State-of-the-Art ............................................... 47
6. Directions in Requirements Traceability ................................... 57
7. Persistent Requirements Traceability Problems .......................... 63
8. Summary of Part I .................................................................. 64

### Part II: Problem

1. Overview of Part II ................................................................. 65
2. Empirical Data Gathering ....................................................... 65
3. Problem Analysis and Definition ............................................. 71
4. Requirements Traceability Framework ..................................... 76
5. Dominating Issues in Pre-Requirements Traceability ................. 82
6. Crux of the Requirements Traceability Problem ....................... 95
7. Supporting Work .................................................................. 103
8. Summary of Part II ............................................................... 111

### Part III: Approach

1. Overview of Part III ............................................................... 112
### Part III: Requirements Engineering

2. Revealing Participants in Requirements Engineering ........................................... 112
3. An Approach Based on Contribution Structures ............................................. 117
4. Relating Agents and Artifacts ........................................................................ 125
5. Developing Contribution Structures .............................................................. 137
6. Relating Artifacts .......................................................................................... 145
7. Modelling Contribution Structures .............................................................. 162
8. Using Contribution Structures ..................................................................... 170
9. Summary of Part III ....................................................................................... 193

### Part IV: Validation

1. Overview of Part IV ......................................................................................... 194
2. Evaluation Method ......................................................................................... 194
3. Tool Support .................................................................................................. 195
4. Tool Description ............................................................................................ 205
5. Scenario of Use ............................................................................................. 219
6. Case Study .................................................................................................... 232
7. Critical Evaluation ......................................................................................... 290
8. Summary of Part IV ....................................................................................... 295

### Part V: Conclusions

1. Overview of Part V ......................................................................................... 296
2. Thesis Argument Revisited .......................................................................... 296
3. Contributions ................................................................................................ 296
4. Outstanding Issues and Future Work ............................................................ 302
5. Technology Transfer ...................................................................................... 309
6. Summary of Part V ....................................................................................... 310
7. Thesis Summary ............................................................................................ 310
Detailed Table of Contents

Abstract 3

Preface 4

i. Statement of Contribution ................................................................. 4
ii. Papers and Reports .............................................................................. 5
iii. Acknowledgements ........................................................................... 6
iv. Financial Support ............................................................................... 7
v. Trademarks .......................................................................................... 7
vi. Disclaimer .......................................................................................... 7
vii. Thesis Road Map.................................................................................. 7

Part I: Introduction .................................................................................. 7
Part II: Problem ....................................................................................... 7
Part III: Approach ................................................................................... 8
Part IV: Validation ................................................................................... 8
Part V: Conclusions ................................................................................ 8

Appendices ............................................................................................... 9

Outline Table of Contents 10

Detailed Table of Contents 12

List of Figures 25
# Detailed Table of Contents

## List of Tables

### Part I: Introduction

1. Overview of Part I ................................................................. 34
2. Thesis Argument .................................................................. 34
   2.1. Issue 34
   2.2. Position ........................................................................ 34
   2.3. Argument ....................................................................... 35
3. Requirements Traceability ..................................................... 35
   3.1. Working Definition of RT .................................................. 35
   3.2. Putting RT into Perspective .............................................. 37
   3.3. Why RT is Important ...................................................... 39
   3.4. Evidence of an RT Problem .............................................. 40
4. State-of-the-Art in Requirements Traceability .......................... 41
   4.1. Review Sources and Organisation .................................... 41
   4.2. Techniques ..................................................................... 42
       4.2.1. Cross Reference-Centred ........................................... 42
       4.2.2. Document-Centred .................................................... 42
       4.2.3. Structure-Centred ...................................................... 43
   4.3. Approaches .................................................................... 43
       4.3.1. Models .................................................................... 43
       4.3.2. Methods .................................................................. 43
       4.3.3. Languages ............................................................... 44
   4.4. Automated Tools .............................................................. 44
       4.4.1. General-Purpose Tools ............................................... 45
       4.4.2. Special-Purpose Tools ............................................... 45
       4.4.3. Workbenches ............................................................ 45
       4.4.4. Environments and Beyond ........................................ 46
### Detailed Table of Contents

5. Critique of the State-of-the-Art ................................................................. 47
  5.1. Priority Given to RT ................................................................. 48
  5.2. Support Provided for RT ........................................................ 48
  5.3. Who Does the RT ................................................................. 49
  5.4. Requirements-Related Information Made Traceable ...................... 50
  5.5. Breadth and Longevity of RT Support ........................................... 51
  5.6. Tasks and Job Roles Supported by RT .......................................... 52
  5.7. Main Strengths ........................................................................... 52
    5.7.1. General-Purpose Tools ....................................................... 53
    5.7.2. Special-Purpose Tools ....................................................... 53
    5.7.3. Workbenches ................................................................... 53
    5.7.4. Environments and Beyond .................................................. 54
  5.8. Main Weaknesses ....................................................................... 54
    5.8.1. General-Purpose Tools ....................................................... 54
    5.8.2. Special-Purpose Tools ....................................................... 55
    5.8.3. Workbenches ................................................................... 55
    5.8.4. Environments and Beyond .................................................. 56
  5.9. Summarising the State-of-the-Art ................................................... 56

6. Directions in Requirements Traceability ................................................. 57
  6.1. Popular Commercial Tools .......................................................... 57
    6.1.1. DOORS ........................................................................... 57
    6.1.2. RTM ............................................................................... 58
    6.1.3. RDD-100 ....................................................................... 59
  6.2. Formal Transformations .................................................................. 61
  6.3. RT Models ................................................................................... 61

7. Persistent Requirements Traceability Problems ...................................... 63
  7.1. Why RT Problems Persist ............................................................ 63
  7.2. A Need for Problem Analysis ......................................................... 64

8. Summary of Part I .............................................................................. 64

**Part II: Problem** 65
# 1. Overview of Part II

- Overview of Part II .......................................................... 65

## 2. Empirical Data Gathering

- Method and Subjects ......................................................... 66
- Surveys and Critiques ....................................................... 68
- Focus Groups ................................................................. 68
- Questionnaires ............................................................... 69
- Interviews ............................................................................ 70
- Observation and Participation ............................................. 70
- Introspection ....................................................................... 71

## 3. Problem Analysis and Definition

- Lack of Common Definition .................................................. 71
  - Purpose-Driven .................................................................... 71
  - Solution-Driven .................................................................... 72
  - Information-Driven ................................................................ 72
  - Direction-Driven ................................................................... 72
  - Implications of Diverse Definitions of RT ........................... 73
- Varied Sources of RT Problems ............................................. 73
  - Implications of Diverse Sources of the RT Problem ............... 74
- Conflicting Needs for RT ...................................................... 74
  - Implications of Diverse Needs for RT ................................. 75
- Multifaceted Nature of the RT Problem .................................. 75

## 4. Requirements Traceability Framework

- Defining RT ......................................................................... 76
- Pre-RT and Post-RT ............................................................ 78
  - Support for Post-RT .......................................................... 79
  - Support for Pre-RT ............................................................ 80
- A Need to Improve Pre-RT .................................................... 80
  - Limited Impact of Improvements in Post-RT on Quality ......... 81
  - Potential Impact of Improvements in Pre-RT on Quality ........ 81
- Pre-RT will be Responsible for Long-Term RT Problems ........... 82

## 5. Dominating Issues in Pre-Requirements Traceability

- Dominating Issues in Pre-Requirements Traceability ............... 82
5.1. Missing Information ........................................................................................................ 83

5.2. Barrier Imposed by a Provider/End-User Conflict ...................................................... 83
   5.2.1. Problems Faced by Providers .................................................................................. 84
   5.2.2. Problems Imposed by End-Users .......................................................................... 86

5.3. Key Requirements for Pre-RT ...................................................................................... 87
   5.3.1. Fundamental Requirements .................................................................................. 88
   5.3.2. Requirements for End-Use ................................................................................... 89
   5.3.3. Requirements for Provision ................................................................................... 89
      5.3.3.1. Obtaining Information ...................................................................................... 89
      5.3.3.2. Recording Information .................................................................................... 90
      5.3.3.3. Organising and Maintaining Information .......................................................... 91

5.4. Addressing Informational Issues .................................................................................. 92
   5.4.1. Increased Awareness of Information to Collect for Pre-RT ......................................... 92
   5.4.2. Obtaining and Recording Information for Pre-RT .................................................... 93
   5.4.3. Organising and Maintaining Information for Pre-RT ............................................... 94
   5.4.4. Flexible Access to and Presentation of Information for Pre-RT ................................. 95

6. Crux of the Requirements Traceability Problem .............................................................. 95
   6.1. Information is Still Missing ........................................................................................ 96
   6.2. Dealing with Missing Information .............................................................................. 98
   6.3. Problems in Practice .................................................................................................. 99
      6.3.1. Causes ................................................................................................................ 99
      6.3.2. Consequences .................................................................................................... 100
      6.3.3. Confirmation ....................................................................................................... 101
   6.4. Lessons from Practice .............................................................................................. 102
   6.5. How can we Tackle the Crux of the RT Problem? ..................................................... 103

7. Supporting Work .............................................................................................................. 103
   7.1. Process Modelling .................................................................................................... 103
      7.1.1. System and Software Process Modelling ................................................................. 104
      7.1.2. Organisational Modelling ................................................................................... 104
   7.2. Project Management Techniques and Tools .............................................................. 106
   7.3. Social Sciences ......................................................................................................... 107
7.4. Document Studies ..............................................................................................................108
7.5. Role Theory ........................................................................................................................109
8. Summary of Part II ....................................................................................................................111

**Part III: Approach**

1. Overview of Part III ..............................................................................................................112
2. Revealing Participants in Requirements Engineering ............................................................112
   2.1. Frequently Asked Questions ............................................................................................113
   2.2. Practical Problems ............................................................................................................114
      2.2.1. Coarse Labels .............................................................................................................114
      2.2.2. Snapshot Labels ........................................................................................................115
   2.3. Political Problems ............................................................................................................116
3. An Approach Based on Contribution Structures ......................................................................117
   3.1. Scope ...............................................................................................................................117
   3.2. Fundamental Requirements .............................................................................................118
   3.3. Terminology and Assumptions .........................................................................................119
      3.3.1. Why RE Artifacts and Agents? .................................................................................119
      3.3.2. Artifacts (Contributions) ..........................................................................................119
      3.3.3. Agents (Contributors) ...............................................................................................120
      3.3.4. Contribution Relations .............................................................................................120
      3.3.5. Contribution Structure .............................................................................................121
      3.3.6. Artifact-Based RT and Personnel-Based RT .............................................................122
      3.3.7. Logical Basis ..............................................................................................................123
   3.4. Steps of the Approach .....................................................................................................124
4. Relating Agents and Artifacts ...............................................................................................125
   4.1. Representing Contribution Relations ................................................................................126
   4.2. Foundations of a Scheme ..................................................................................................126
      4.2.1. Goffman's Work .......................................................................................................127
      4.2.2. Other Related Work ..................................................................................................129
   4.3. Contribution Format .........................................................................................................130
      4.3.1. Principal ..................................................................................................................131
4.3.2. Author ........................................................................................................... 131
4.3.3. Documentor .................................................................................................. 131
4.3.4. Layers ........................................................................................................... 131
4.4. Qualification ...................................................................................................... 132
  4.4.1. Of Principal ................................................................................................. 133
  4.4.2. Of Author .................................................................................................... 134
  4.4.3. Of Documentor ............................................................................................ 134
4.5. Example of a Rudimentary Contribution Structure .......................................... 135

5. Developing Contribution Structures ..................................................................... 137
  5.1. Uncovering Hidden Details .......................................................................... 137
  5.2. Foundations for Further Analyses .................................................................. 138
  5.3. Social Roles and Role Relations .................................................................... 139
    5.3.1. Implications .............................................................................................. 141
  5.4. Commitment ................................................................................................... 142
    5.4.1. Individual and Collective Commitment .................................................... 143
    5.4.2. Social Commitment .................................................................................. 143
    5.4.3. Implications .............................................................................................. 144
  5.5. Example of Further Analyses ........................................................................ 144

6. Relating Artifacts ................................................................................................ 145
  6.1. Accounting for Artifact-Based RT Relations .................................................. 146
  6.2. Categories of Artifact-Based RT Relation ...................................................... 146
    6.2.1. Temporal Relations .................................................................................. 147
    6.2.2. Developmental Relations ......................................................................... 149
    6.2.3. Auxiliary Relations .................................................................................. 151
  6.3. Containment Relations .................................................................................... 152
    6.3.1. Example of Use ....................................................................................... 153
  6.4. Connectivity Relations .................................................................................... 154
    6.4.1. Linguistic Basis ......................................................................................... 154
    6.4.2. Working Set of Relations ......................................................................... 155
      6.4.2.1. References Connectivity Relation ...................................................... 156
      6.4.2.2. Adopts Connectivity Relation .............................................................. 157
## Detailed Table of Contents

6.4.3. Purposes and Qualifiers ......................................................... 158  
6.4.4. Example of Use .......................................................................... 159  
6.5. Example of Auxiliary Relations in Use........................................ 161  
7. Modelling Contribution Structures .................................................. 162  
  7.1. Model Preliminaries ........................................................................ 162  
    7.1.1. Basic Types ............................................................................. 162  
    7.1.2. Type Definitions ..................................................................... 162  
    7.1.3. Derived Types ........................................................................ 163  
    7.1.4. Specification of State ............................................................... 164  
    7.1.5. Specification of Initial State ..................................................... 165  
    7.1.6. Invariants ............................................................................... 165  
  7.2. Specification of Operations to Alter State ..................................... 166  
    7.2.1. Adding Artifacts, Agents and Relations ................................. 167  
    7.2.2. Deleting Artifacts, Agents and Relations ................................. 168  
8. Using Contribution Structures............................................................ 170  
  8.1. Notes on Notation .......................................................................... 171  
    8.1.1. Predicates ............................................................................... 172  
    8.1.2. Constructs .............................................................................. 172  
  8.2. Rudimentary True/False Query Operations .................................. 173  
    8.2.1. Agent/Artifact-Centred Examples ......................................... 173  
    8.2.2. Artifact-Centred Examples .................................................... 175  
    8.2.3. Agent-Centred Examples ....................................................... 177  
  8.3. Rudimentary Construct Query Operations ................................... 180  
    8.3.1. Agent/Artifact-Centred Examples ......................................... 180  
    8.3.2. Artifact-Centred Examples .................................................... 183  
    8.3.3. Agent-Centred Examples ....................................................... 185  
  8.4. Towards More Complex Query Operations ................................... 187  
  8.5. Some Answers to Frequently Asked Questions ............................... 187  
    8.5.1. FAQ 1: Involvement ................................................................. 188  
    8.5.2. FAQ 2: Responsibility ............................................................. 188  
    8.5.3. FAQ 3: Working Arrangement ............................................... 189
## Detailed Table of Contents

- 8.5.4. FAQ 4: Change Notification .................................................. 189  
- 8.5.5. FAQ 5: Ramification ............................................................. 190  
- 8.6. Towards Further Analyses ...................................................... 190  
- 9. Summary of Part III .................................................................. 193  

### Part IV: Validation

1. Overview of Part IV .................................................................... 194  
2. Evaluation Method ..................................................................... 194  
3. Tool Support .............................................................................. 195  
   3.1. Objectives and Scope ............................................................ 196  
   3.2. High-Level Requirements .................................................... 197  
   3.3. Task Model ......................................................................... 197  
      3.3.1. A Role-Oriented Task Model ........................................... 198  
      3.3.2. Task Model for Providers ............................................. 200  
      3.3.3. Task Model for End-Users ............................................ 202  
   3.4. Assumptions and Decisions .................................................. 204  
4. Tool Description ........................................................................ 205  
   4.1. Overall Architecture and Implementation ............................. 205  
   4.2. Traceability Extension Tool ................................................... 206  
      4.2.1. Project and Work Spaces ............................................. 207  
      4.2.2. RT Markup Language ................................................... 208  
   4.3. Contribution Structure Manager ......................................... 213  
      4.3.1. Modelling and Exporting Contribution Structures .......... 213  
      4.3.2. Using Contribution Structures ....................................... 214  
   4.4. Considerations for a Production Version ............................... 218  
5. Scenario of Use ......................................................................... 219  
   5.1. Description of Scenario ....................................................... 219  
   5.2. How the Approach Represents the Scenario ....................... 221  
   5.3. What the Approach Brings to the Scenario ......................... 226  
6. Case Study ................................................................................. 232  
   6.1. Description of the Case Study .............................................. 232
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1. Company and Project</td>
<td>232</td>
</tr>
<tr>
<td>6.1.2. Data Gathered</td>
<td>233</td>
</tr>
<tr>
<td>6.1.3. Agents</td>
<td>234</td>
</tr>
<tr>
<td>6.1.4. Artifacts</td>
<td>234</td>
</tr>
<tr>
<td>6.2. Application of the Approach</td>
<td>235</td>
</tr>
<tr>
<td>6.2.1. Artifact-Based RT Relations</td>
<td>236</td>
</tr>
<tr>
<td>6.2.2. Contribution Format</td>
<td>246</td>
</tr>
<tr>
<td>6.2.3. Qualification of Contribution Format</td>
<td>248</td>
</tr>
<tr>
<td>6.2.4. Social Roles and Role Relations</td>
<td>252</td>
</tr>
<tr>
<td>6.2.5. Agent Commitments to Artifacts and to Each Other</td>
<td>254</td>
</tr>
<tr>
<td>6.3. Personnel-Based RT and the Frequently Asked Questions</td>
<td>255</td>
</tr>
<tr>
<td>6.3.1. FAQ 1: Involvement</td>
<td>256</td>
</tr>
<tr>
<td>6.3.2. FAQ 2: Responsibility</td>
<td>258</td>
</tr>
<tr>
<td>6.3.3. FAQ 3: Working Arrangement</td>
<td>259</td>
</tr>
<tr>
<td>6.3.4. FAQ 4: Change Notification</td>
<td>260</td>
</tr>
<tr>
<td>6.3.5. FAQ 5: Ramification</td>
<td>262</td>
</tr>
<tr>
<td>6.3.6. FAQ 6: Source</td>
<td>265</td>
</tr>
<tr>
<td>6.3.7. FAQ 7: Remit</td>
<td>266</td>
</tr>
<tr>
<td>6.3.8. FAQ 8: Copying</td>
<td>267</td>
</tr>
<tr>
<td>6.3.9. FAQ 9: Use</td>
<td>268</td>
</tr>
<tr>
<td>6.3.10. FAQ 10: Collaboration</td>
<td>269</td>
</tr>
<tr>
<td>6.4. Further Analyses</td>
<td>271</td>
</tr>
<tr>
<td>6.4.1. Number of Contributors to Artifacts</td>
<td>271</td>
</tr>
<tr>
<td>6.4.2. Number of Contributors to Artifacts in the Three Capacities</td>
<td>276</td>
</tr>
<tr>
<td>6.4.3. Number and Type of Contributions by Specific Agents</td>
<td>278</td>
</tr>
<tr>
<td>6.4.4. Key Contributors</td>
<td>283</td>
</tr>
<tr>
<td>6.4. Discussion</td>
<td>286</td>
</tr>
<tr>
<td>6.5.1. Issues</td>
<td>286</td>
</tr>
<tr>
<td>6.5.2. Implications for Practice</td>
<td>289</td>
</tr>
<tr>
<td>7. Critical Evaluation</td>
<td>290</td>
</tr>
<tr>
<td>7.1. Evaluation Framework</td>
<td>290</td>
</tr>
</tbody>
</table>
### Part V: Conclusions

1. Overview of Part V ................................................................. 296
2. Thesis Argument Revisited .................................................. 296
3. Contributions .................................................................... 297
   3.1. To the Discipline of Software Engineering ....................... 297
   3.2. Surveying the State-of-the-Art in RT Research and Practice .... 299
   3.3. Understanding RT and the So-Called "RT problem" ............... 299
   3.4. Uncovering the Need to Focus Effort on the Issues of Pre-RT ... 299
   3.5. Identifying the Crux of the RT Problem ............................ 300
   3.6. Introducing Contribution Structures and Personnel-Based RT ... 300
   3.7. Providing Additional and Far-Reaching Consequences .......... 300
      3.7.1. Firmest Foundations for Quality Development ............... 301
      3.7.2. Firmer Foundations for Contemporary Analyses ............. 301
      3.7.3. Foundations for RE Process Improvement ................... 302
4. Outstanding Issues and Future Work ........................................ 302
   4.1. Incomplete Coverage ...................................................... 303
      4.1.1. Accounting for Indirect Contributors and Recipients ...... 303
      4.1.2. Accounting for Evolving Requirements ....................... 304
      4.1.3. Accounting for Discussion About Requirements ............. 304
   4.2. Preliminary Categories ................................................... 305
      4.2.1. Accounting for Actual Contribution Relations and Social Roles 305
### Detailed Table of Contents

- **4.2.2. Accounting for Actual Artifacts and Artifact-Based Relations** .................................................. 305
- **4.2.3. Accounting for Types of Requirements Change** ................................................................. 306
- **4.3. Organisational Resistance** ............................................................................................................ 306
  - **4.3.1. Changing Organisational Policies** .......................................................................................... 306
  - **4.3.2. Automatically Capturing the Information Needed** ................................................................. 307
  - **4.3.3. Instantiation in and Integration with Third-Party Tools** ......................................................... 307
- **4.4. Implementation Issues** .................................................................................................................. 308
  - **4.4.1. Dealing with Physical and Conceptual Distribution** .............................................................. 308
  - **4.4.2. Improving RT Visualisation and Navigation** ......................................................................... 308
  - **4.4.3. Determining Scaleability and Generalisability** .................................................................. 308
- **5. Technology Transfer** ....................................................................................................................... 309
- **6. Summary of Part V** ......................................................................................................................... 310
- **7. Thesis Summary** ............................................................................................................................. 310

### Glossary

**References** ............................................................................................................................................ 314

### Appendices

**Appendices** ............................................................................................................................................ See separate volume

- **A. Empirical Work** .................................................................................................................................. 14
  - **A.1. Focus Groups** ............................................................................................................................ 14
  - **A.2. About the Two-Stage Questionnaire** ....................................................................................... 45
  - **A.3. Questionnaire One** .................................................................................................................. 47
  - **A.4. Questionnaire Two** ................................................................................................................... 67
- **B. More Modelling** .................................................................................................................................. 156
  - **B.1. Predicates** ............................................................................................................................... 156
  - **B.2. Sets** .......................................................................................................................................... 175
- **C. Tool-Related Documentation** ......................................................................................................... 203
  - **C.1. Requirements Specification for Pre-Requirements Traceability** .............................................. 203
  - **C.2. Requirements Specification for Contribution Structures** ....................................................... 281
  - **C.3. Task Model for Contribution Structures** .................................................................................. 308
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>D. Case Study Material</td>
<td>329</td>
</tr>
<tr>
<td>D.1. Agents</td>
<td>329</td>
</tr>
<tr>
<td>D.2. Artifacts</td>
<td>332</td>
</tr>
<tr>
<td>D.3. Artifact-Based RT Relations</td>
<td>340</td>
</tr>
<tr>
<td>D.4. Contribution Formats</td>
<td>362</td>
</tr>
<tr>
<td>D.5. Agent Contributions</td>
<td>378</td>
</tr>
</tbody>
</table>
List of Figures

Figure 1: The distinction between horizontal, vertical, forwards, and backwards RT ............37
Figure 2: The various data gathering techniques used to collect data about the RT problem........................................................................................................................................67
Figure 3: Decomposition of one of the problems involved in RT ........................................69
Figure 4: A simplified diagram to show the two basic types of RT ........................................79
Figure 5: The main problems underlying the provision of pre-RT ........................................85
Figure 6: The main problems imposed by the end-users of pre-RT ........................................87
Figure 7: A basic responsibility matrix used in project management .....................................107
Figure 8: A basic matrix used to form citation networks ........................................................108
Figure 9: Banton's role tree .....................................................................................................110
Figure 10: Typical coarse-grained labelling conventions .......................................................115
Figure 11: How changing authorship details are typically accounted for .............................116
Figure 12: A coarse-grained contribution relation ..................................................................121
Figure 13: Finer-grained contribution relations ......................................................................121
Figure 14: The contribution structure underlying RE ...........................................................122
Figure 15: Artifact-based RT and personnel-based RT .......................................................123
Figure 16: The main steps of the approach ..........................................................................125
Figure 17: A flat and coarse model of the contribution structure ..........................................126
Figure 18: Goffman's decomposition of participant roles .....................................................128
Figure 19: The three capacities which define the contribution format of an artifact.............131
Figure 20: Using three contribution layers to organise the model of contribution structure ......................................................................................................................132
Figure 21: Example attributes to qualify the three capacities of the contribution format ....133
List of Figures

Figure 22: The contribution format of a primitive and isolated artifact defines its contribution structure .......................................................... 136
Figure 23: The contribution formats of all its internal components and their interrelations defines the contribution structure of a composite artifact .......... 136
Figure 24: An example set of social roles ................................................................. 140
Figure 25: Using seven contribution layers to organise the model of contribution structure ........................................................................ 141
Figure 26: How some overloaded terminology is being used in this thesis .................. 142
Figure 27: A devisor/relayer role relation .................................................................. 145
Figure 28: Historical structure and historical RT ..................................................... 148
Figure 29: Developmental structure and developmental RT .................................... 150
Figure 30: Auxiliary structures and forms of RT ...................................................... 152
Figure 31: The layered contribution formats of a composite artifact ......................... 153
Figure 32: Seven conditions that must be met for textuality ..................................... 155
Figure 33: The "references" connectivity relation ..................................................... 156
Figure 34: The "adopts" connectivity relation .......................................................... 158
Figure 35: Identifying connectivity relations ........................................................... 159
Figure 36: How connectivity relations qualify the mode of authorship ..................... 160
Figure 37: Containment and connectivity relations in use ........................................ 161
Figure 38: Examples of some rudimentary query operations which can be used to clarify artifact-based RT ........................................................................ 171
Figure 39: The general form of query operations in our modelling scheme ................. 171
Figure 40: Agent/artifact-centred examples .................................................................. 174
Figure 41: Artifact-centred examples ........................................................................ 175
Figure 42: Agent-centred examples ........................................................................... 178
Figure 43: Further agent/artifact-centred examples ................................................. 180
Figure 44: Further artifact-centred examples .......................................................... 183
Figure 45: Further agent-centred examples .............................................................. 185
Figure 46: The strategy adopted for our prototyping exercise ................................... 196
Figure 47: An example of some basic operations and their sequencing when modelling and using contribution structures ........................................... 199
Figure 48: The first sub-goal, SG-A1, of the main goal for providers ........................................201
Figure 49: The second sub-goal, SG-A2, of the main goal for providers ........................................202
Figure 50: The first sub-goal, SG-B1, of the main goal for end-users ........................................203
Figure 51: The second sub-goal, SG-B2, of the main goal for end-users ........................................204
Figure 52: The architecture of the tool ..........................................................................................206
Figure 53: An empty workspace in the traceability extension tool ..............................................207
Figure 54: The basic menu options from the traceability extension tool ........................................208
Figure 55: Some primitives of the RT markup language .................................................................209
Figure 56: View of an artifact without its RT markup .................................................................210
Figure 57: Underlying view of an artifact with its RT markup .....................................................210
Figure 58: An internal component of an artifact and its contribution format .................................212
Figure 59: Markup-related menu options from the traceability extension tool ..............................213
Figure 60: The details provided in an artifact profile .................................................................215
Figure 61: The details provided in an agent profile ......................................................................215
Figure 62: Query-related menu options from the traceability extension tool ...............................216
Figure 63: Reporting on a forwards RT request .........................................................................217
Figure 64: Reporting on the various relations between specified artifacts .................................217
Figure 65: Reporting on the various relations between specified agents ......................................218
Figure 66: The main steps of the approach ..................................................................................221
Figure 67: The historical order of artifact production in the scenario .........................................222
Figure 68: The developmental order of artifact production in the scenario ..................................223
Figure 69: The auxiliary relations and contribution formats in the scenario .................................224
Figure 70: How an artifact instigating change can be deduced from connectivity relations ..........225
Figure 71: Trace to the original artifact source of paragraph y in the scenario, with the associated contribution formats retrieved along the way ...........................................227
Figure 72: Artifact profiles for the initial RS, revised RS, paragraph x and paragraph y of the scenario ..................................................................................................................228
Figure 73: Agent profiles for RE member 2, project leader, MD and user 1 of the scenario .............229
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>74</td>
<td>Some details about the change from paragraph x to paragraph y in the scenario.</td>
</tr>
<tr>
<td>75</td>
<td>Following up change details by looking at the relations between the agents concerned in the scenario.</td>
</tr>
<tr>
<td>76</td>
<td>Following up change details by looking at the artifact that instigated the change in the scenario.</td>
</tr>
<tr>
<td>77</td>
<td>The order in which artifacts 1 to 23 were produced and the predominant artifact-based RT relations between them.</td>
</tr>
<tr>
<td>78</td>
<td>The order in which artifacts 24 to 88 were produced and the predominant artifact-based RT relations between them.</td>
</tr>
<tr>
<td>79</td>
<td>The order in which artifacts 89 to 127 were produced and the predominant artifact-based RT relations between them.</td>
</tr>
<tr>
<td>80</td>
<td>The order in which artifacts 128 to 166 were produced and the predominant artifact-based RT relations between them.</td>
</tr>
<tr>
<td>81</td>
<td>The temporal relations between artifacts 1 to 23.</td>
</tr>
<tr>
<td>82</td>
<td>Those artifact-based RT relations that are likely to be maintained between artifacts 1 to 23 by conventional RT schemes.</td>
</tr>
<tr>
<td>83</td>
<td>Those further artifact-based RT relations that can be brought into the picture between artifacts 1 to 23.</td>
</tr>
<tr>
<td>84</td>
<td>Authorial trail to artifact 106.</td>
</tr>
<tr>
<td>85</td>
<td>Continuation of an internal authorial trail in Figure 84.</td>
</tr>
<tr>
<td>86</td>
<td>Authorial trail from artifact 106.</td>
</tr>
<tr>
<td>87</td>
<td>Who was involved in the production of security requirements.</td>
</tr>
<tr>
<td>88</td>
<td>Those agents that need to be involved in, and informed of, any changes proposed to the section on fax mailbox services in the operations manual.</td>
</tr>
<tr>
<td>89</td>
<td>Those agents that can provide more information about the call divert service.</td>
</tr>
<tr>
<td>90</td>
<td>The changing decision making authority amongst members of the key group contributing to the requirements and design specification.</td>
</tr>
<tr>
<td>91</td>
<td>Those agents who used agent AG's marketing plan and how.</td>
</tr>
<tr>
<td>92</td>
<td>The number of contributors to the artifacts produced in phase one.</td>
</tr>
<tr>
<td>93</td>
<td>The number of contributors to the artifacts produced in phase two.</td>
</tr>
<tr>
<td>94</td>
<td>The number of contributors to the artifacts produced in phase three.</td>
</tr>
<tr>
<td>95</td>
<td>The number of contributors to the artifacts produced in phase four.</td>
</tr>
</tbody>
</table>
Figure 96: The number of contributors to all the artifacts produced ........................................276
Figure 97: The number of contributors in each capacity to the artifacts produced in phase one .........................................................................................................................277
Figure 98: The total number of artifacts different agents have contributed to .........................279
Figure 99: The total number of artifacts different agents have contributed to as principal ......279
Figure 100: The total number of artifacts different agents have contributed to as author ........280
Figure 101: The total number of artifacts different agents have contributed to as documentor ........................................................................................................................................280
Figure 102: The contributor profile of agent AE ........................................................................281
Figure 103: The contributor profile of agent AA ........................................................................282
Figure 104: The percentage of agent AE's contributions in each of the three capacities ..........282
Figure 105: The percentage of agent AA's contributions in each of the three capacities ........283
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>The main strengths of the different classes of tool support for RT</td>
<td>53</td>
</tr>
<tr>
<td>Table 2</td>
<td>The main weaknesses of the different classes of tool support for RT</td>
<td>54</td>
</tr>
<tr>
<td>Table 3</td>
<td>Fundamental requirements for pre-RT</td>
<td>88</td>
</tr>
<tr>
<td>Table 4</td>
<td>Basic requirements for end-use of pre-RT</td>
<td>89</td>
</tr>
<tr>
<td>Table 5</td>
<td>Basic requirements for obtaining information to provide pre-RT</td>
<td>90</td>
</tr>
<tr>
<td>Table 6</td>
<td>Basic requirements for recording information to provide pre-RT</td>
<td>90</td>
</tr>
<tr>
<td>Table 7</td>
<td>Basic requirements for organising and maintaining information to provide</td>
<td>91-92</td>
</tr>
<tr>
<td></td>
<td>pre-RT</td>
<td></td>
</tr>
<tr>
<td>Table 8</td>
<td>A partitioning of role types</td>
<td>110</td>
</tr>
<tr>
<td>Table 9</td>
<td>Partitioning the social dimension of RE</td>
<td>130</td>
</tr>
<tr>
<td>Table 10</td>
<td>Derived production roles or social roles</td>
<td>139</td>
</tr>
<tr>
<td>Table 11</td>
<td>An example set of rudimentary commitments to artifacts</td>
<td>143</td>
</tr>
<tr>
<td>Table 12</td>
<td>Decomposition of the &quot;references&quot; group of connectivity relations</td>
<td>157</td>
</tr>
<tr>
<td>Table 13</td>
<td>Decomposition of the &quot;adopts&quot; group of connectivity relations</td>
<td>158</td>
</tr>
<tr>
<td>Table 14</td>
<td>An incidence matrix indicating the number of relations between artifacts</td>
<td>191</td>
</tr>
<tr>
<td>Table 15</td>
<td>An incidence matrix indicating the number of relations between agents</td>
<td>191</td>
</tr>
<tr>
<td>Table 16</td>
<td>An adjacency matrix indicating contribution relations between agents and</td>
<td>191</td>
</tr>
<tr>
<td></td>
<td>artifacts</td>
<td></td>
</tr>
<tr>
<td>Table 17</td>
<td>Progressively detailed semantics for the main artifact-based RT relations</td>
<td>241-242</td>
</tr>
<tr>
<td></td>
<td>between artifacts 1 to 23</td>
<td></td>
</tr>
<tr>
<td>Table 18</td>
<td>The contribution formats for artifacts 1 to 23</td>
<td>247</td>
</tr>
<tr>
<td>Table 19</td>
<td>Contributors to artifacts 1 to 23</td>
<td>248</td>
</tr>
<tr>
<td>Table 20</td>
<td>The social roles and role relations for agent AI</td>
<td>252</td>
</tr>
<tr>
<td>Table 21</td>
<td>The social roles and role relations for agent AJ</td>
<td>253</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>------------------------------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>Table 22</td>
<td>Agent AP’s artifact commitment store</td>
<td>254</td>
</tr>
<tr>
<td>Table 23</td>
<td>Agent AP’s agent commitment store</td>
<td>255</td>
</tr>
<tr>
<td>Table 24</td>
<td>Changes in responsibility for the manager’s guide</td>
<td>258</td>
</tr>
<tr>
<td>Table 25</td>
<td>The working arrangements of those involved in the versions of the requirements and design specification</td>
<td>260</td>
</tr>
<tr>
<td>Table 26</td>
<td>Agent AC’s contributions, collaborators, and the subsequent project artifacts which have made use of his contributions in some way</td>
<td>263-264</td>
</tr>
<tr>
<td>Table 27</td>
<td>The collaborations of agents AE and AA in phase one</td>
<td>269</td>
</tr>
<tr>
<td>Table 28</td>
<td>The collaborations of agents AE and AA in phase two</td>
<td>270</td>
</tr>
<tr>
<td>Table 29</td>
<td>The collaborations of agents AE and AA in phase three</td>
<td>270</td>
</tr>
<tr>
<td>Table 30</td>
<td>Agent AE’s social roles when he collaborated with agent AA</td>
<td>270</td>
</tr>
<tr>
<td>Table 31</td>
<td>Agent AA’s social roles when he collaborated with agent AE</td>
<td>271</td>
</tr>
<tr>
<td>Table 32</td>
<td>Summary information about project contributors</td>
<td>278</td>
</tr>
<tr>
<td>Table 33</td>
<td>The key contributors to the DR project</td>
<td>283</td>
</tr>
<tr>
<td>Table 34</td>
<td>The phases of the DR project in which the key contributors have contributed</td>
<td>284</td>
</tr>
<tr>
<td>Table 35</td>
<td>The number of artifacts on which the key contributors have collaborated</td>
<td>285</td>
</tr>
<tr>
<td>Table 36</td>
<td>With how many other agents key agents collaborated when they contributed</td>
<td>286</td>
</tr>
<tr>
<td>Table 37</td>
<td>Summary of outstanding issues and future work</td>
<td>303</td>
</tr>
<tr>
<td>Table 38</td>
<td>Partitioning the social dimension of RE</td>
<td>304</td>
</tr>
<tr>
<td>Box 1:</td>
<td>A working definition of the term &quot;requirements traceability&quot;</td>
<td>35</td>
</tr>
<tr>
<td>-------</td>
<td>-------------------------------------------------------------</td>
<td>----</td>
</tr>
<tr>
<td>Box 2:</td>
<td>A definition of the term &quot;requirement&quot;</td>
<td>36</td>
</tr>
<tr>
<td>Box 3:</td>
<td>Definitions of the term &quot;requirements engineering&quot;</td>
<td>38</td>
</tr>
<tr>
<td>Box 4:</td>
<td>A definition of the term &quot;requirement specification&quot;</td>
<td>39</td>
</tr>
<tr>
<td>Box 5:</td>
<td>A definition often used for the term &quot;requirements traceability&quot;</td>
<td>76</td>
</tr>
<tr>
<td>Box 6:</td>
<td>What is meant by the word &quot;trace&quot;</td>
<td>77</td>
</tr>
<tr>
<td>Box 7:</td>
<td>A working definition of the term &quot;requirements traceability&quot;</td>
<td>77</td>
</tr>
<tr>
<td>Box 8:</td>
<td>A working definition of the term &quot;pre-requirements traceability&quot;</td>
<td>78</td>
</tr>
<tr>
<td>Box 9:</td>
<td>A working definition of the term &quot;post-requirements traceability&quot;</td>
<td>78</td>
</tr>
<tr>
<td>Box 10:</td>
<td>The notions of &quot;role&quot; and &quot;role relation&quot; are central to the analysis of social situations</td>
<td>109</td>
</tr>
<tr>
<td>Box 11:</td>
<td>A definition of the term &quot;social structure&quot;</td>
<td>113</td>
</tr>
<tr>
<td>Box 12:</td>
<td>How behaviour binds an individual to an organisation</td>
<td>124</td>
</tr>
<tr>
<td>Box 13:</td>
<td>Describing the multiple and changing nature of social relations</td>
<td>138</td>
</tr>
<tr>
<td>Box 14:</td>
<td>An explanation of the term &quot;intertextuality&quot;</td>
<td>146</td>
</tr>
<tr>
<td>Box 15:</td>
<td>Question 1 of the frequently asked questions - Involvement</td>
<td>188</td>
</tr>
<tr>
<td>Box 16:</td>
<td>Question 2 of the frequently asked questions - Responsibility</td>
<td>188</td>
</tr>
<tr>
<td>Box 17:</td>
<td>Question 3 of the frequently asked questions - Working Arrangement</td>
<td>189</td>
</tr>
<tr>
<td>Box 18:</td>
<td>Question 4 of the frequently asked questions - Change Notification</td>
<td>189</td>
</tr>
<tr>
<td>Box 19:</td>
<td>Question 5 of the frequently asked questions - Ramification</td>
<td>190</td>
</tr>
<tr>
<td>Box 20:</td>
<td>The basic concepts used to describe the role-oriented task model</td>
<td>200</td>
</tr>
<tr>
<td>Box 21:</td>
<td>Scenario text</td>
<td>220</td>
</tr>
<tr>
<td>Box 22:</td>
<td>Question 1 of the frequently asked questions - Involvement</td>
<td>256</td>
</tr>
</tbody>
</table>
### List of Boxes

| Box 23 | Question 2 of the frequently asked questions - Responsibility | 258 |
| Box 24 | Question 3 of the frequently asked questions - Working Arrangement | 259 |
| Box 25 | Question 4 of the frequently asked questions - Change Notification | 260 |
| Box 26 | Question 5 of the frequently asked questions - Ramification | 262 |
| Box 27 | Question 6 of the frequently asked questions - Source | 265 |
| Box 28 | Question 7 of the frequently asked questions - Remit | 266 |
| Box 29 | Question 8 of the frequently asked questions - Copying | 267 |
| Box 30 | Question 9 of the frequently asked questions - Use | 268 |
| Box 31 | Question 10 of the frequently asked questions - Collaboration | 269 |
| Box 32 | A definition of the term "software engineering" | 298 |
| Box 33 | Definitions of the sub-fields of software engineering in which we have contributed | 298 |
Part I: Introduction

1. Overview of Part I

In Part I, we begin with a description of the thesis argument. We then introduce the broad topic of requirements traceability (RT) with which this thesis is concerned. This material consists of some preliminary definitions and background information which locates the topic of RT within the discipline of systems and software engineering in general and within the discipline of requirements engineering (RE) in particular. It explains why RT is important, points to evidence of a problem with current RT practice, and so indicates why it is a critical problem to address. We then critique the state-of-the-art in RT support, followed by a report on the direction of contemporary research and practice. In the light of this material, we examine why an RT problem still persists, and conclude the need for a thorough analysis of this problem. We argue that it is essential to understand what the RT problem actually is if we are to know what a potential solution would have to address.

2. Thesis Argument

In this section, we lay out the structure of the thesis argument. To do this, we provide a statement of the problem which is at issue, a statement of the position we maintain with respect to this issue, and a summary as to how we argue and justify this position.

2.1. Issue

The problem that we are interested in is that of RT. This is because RT is currently cited as one of the dominating problem areas confronting systems and software engineering practice. Our objective is to examine how experience of RT problems can be reduced in practice.

2.2. Position

We suggest that the so-called "RT problem" is multifaceted in nature and unlikely to have an all-encompassing solution. Therefore, the position we maintain in this thesis is that the greatest
potential for reducing RT problems, especially in the longer-term, lies in addressing those fundamental issues which underlie extant problems.

2.3. Argument

We corroborate the above position in the following manner:

(1) Conducting a problem analysis to locate the crux of the RT problem.

(2) Developing an approach to tackle the crux of the RT problem.

(3) Validating that the approach provides both a practical and feasible solution to the crux of the RT problem.

(4) Demonstrating how the approach provides a way to directly reduce many experiences of RT problems and to indirectly alleviate many others.

3. Requirements Traceability

In this section, we introduce the topic of RT. We first provide definitions for some of the terminology we use. We then place RT in its historical context and consider reasons for a recent growth of interest in the area. We follow this with a general explanation as to why RT is crucial for improving the discipline of systems and software engineering and, in particular, for improving RE. We then point out that, although improvements in RT have been forthcoming, industry still continues to be plagued by RT problems.

3.1. Working Definition of RT

Definitions of the term "requirements traceability" are discussed in detail in Part II of this thesis. However, our working definition of this term is provided in Box 1 for reader orientation.

"Requirements traceability (RT) refers to the ability to describe and follow the life of a requirement in both a forwards and backwards direction (i.e., from its origins, through its development and specification, to its subsequent deployment and use, and through periods of on-going refinement and iteration in any of these phases)."

**Box 1:** A working definition of the term "requirements traceability" (taken from [Gotel & Finkelstein 1994]).

Box 2 provides a definition of the term "requirement". Note that in this thesis we are not concerned with the precise nature of requirements and so treat all requirements in a homogeneous
manner.

"Requirement - In system/software engineering:

1. A capability needed by a user to solve a problem or achieve an objective.

2. A capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document.

3. The set of all requirements that form the basis for subsequent development of the software or software component.

4. Short description sometimes used in place of the term software requirements specification."

Box 2: A definition of the term "requirement" (taken from [Thayer & Dorfman 1990]).

We later make use of the terms "horizontal RT" and "vertical RT". These refer to the phases of a requirement's life through which RT is carried out. Horizontal RT is traceability between versions and variants of a requirement within a particular phase of its life. It provides breadth of RT coverage. Vertical RT is traceability between previous or subsequent phases of a requirement's life. It provides depth of RT coverage. The terms "forwards RT" and "backwards RT" refer to the direction in which RT is carried out. If the RT is vertical, forwards RT is traceability from the origin of a requirement towards its realisation, whereas backwards RT is traceability from the realisation of a requirement towards its origin. These distinctions are illustrated in Figure 1.
3.2. Putting RT into Perspective

Alford reports how the practice of RT has been going on, in one form or another, for at least forty years [Alford 1994b]. In this brief article, he explains why RT originated with the introduction of systems engineering in the 1950’s, how it became operational when the focus turned to software engineering in the 1970’s, and further highlights the ways in which it has evolved over the subsequent years. Throughout this period, although there have been repeated calls to improve RT practice, seen from [Alford 1977] through to [Fickas & Finkelstein 1993], both industrial and academic interest in RT has never before been quite so visible as it is today.

One reason for the current high profile of RT is the recent surge of interest in RE. This is because RT is an integral aspect of RE. Two popular definitions of the term "requirements engineering" are given in Box 3, again for reader orientation, and are individually discussed no further.
Part I: Introduction

"Requirements engineering is the science and discipline concerned with analyzing and documenting requirements, including needs analysis, requirements analysis, and requirements specification. It also provides the appropriate mechanisms to facilitate the analysis, documentation, and verification activities."

"Requirements engineering is the branch of software engineering concerned with the real-world goals for, functions of, and constraints on software systems. It is also concerned with the relationship of these factors to precise specifications of software behavior, and to their evolution over time and across software families."

**Box 3: Definitions of the term "requirements engineering" (taken from [Thayer & Dorfman 1990] and [Zave 1995] respectively).**

Prior to the 1990's, work in RE appears somewhat disjoint and dispersed in the literature. Notable exceptions include a special issue of a popular journal dedicated to RE [IEEE 1977] and the proceedings of an international symposium on RE environments [Ohno 1982]. Today, RE is a vibrant and rapidly growing field. This could be said to be a consequence of issues like:

1. A switch in focus to the upstream aspects of systems and software development as the issues related to the downstream aspects become more competently tackled using structured methods and computer support [Sodhi 1991].

2. A growing acceptance of the early arguments put forward concerning the greater leverage that can be gained by tackling those issues which cause problems in the upstream phases of the systems and software lifecycle [Boehm 1981].

3. Periodic reports providing a rather negative assessment of the state-of-the-art in RE and calls for the need to establish an RE community [Davis & Freeman 1991].

4. A trend towards software outsourcing in industrial organisations [Jones 1994].


The goal of RE is to produce a baseline description from which systems and software can be built, and against which the development process and its final outcome can be assessed. Such a description aims to document all the essential requirements of the system. This is commonly referred to as the *requirements specification* (RS) and is defined in Box 4. The manner in which
these requirements are determined is a primary concern of RE, the subject of much debate, and an area we do not pursue here. Similarly, the exact content and structure of an RS is also a controversial issue, though there is some agreement as to the attributes that any RS should exhibit [Davis 1990, Roman 1985, Ross & Schoman 1977, Sommerville 1989]. Common to all such lists is that an RS be traceable. This, in turn, implies that the requirements in the RS be traceable. Establishing the potential for RT is not only a major sub-goal of RE, but RT is further crucial for conducting the RE process itself.

"Requirement specification - In system/software engineering, a document that states the functions that software must perform, required level of performance (speed, accuracy, etc.), nature of the required interfaces between the software product and its environment, the type and severity of constraints on design, and the quality of the final product."

Box 4: A definition of the term "requirement specification" (taken from [Thayer & Dorfman 1990]).

3.3. Why RT is Important

Definitions of the term "quality" within the literature suggest that system or software quality is assured if: (a) the system or software meets its users' needs, its user requirements; and/or (b) the system or software conforms to rigid quality attributes, its quality requirements. We claim that RT is important as it provides the primary mechanism through which quality systems and software can be engineered and delivered.

Today, systems and software continue to be built which neither conform to specified quality standards, nor meet user needs on delivery. This situation can be said to be due to a combination of factors like:

1. Inadequate requirements elicitation.
2. The inability to transcribe elicited requirements into a tangible or representative form.
3. The difficulty in reconciling diverse requirements.
4. The different interpretations given to requirements throughout their development and use.
5. Having to deal with additional and changing requirements.

Although improvements in elicitation, negotiation, and description can offer more assurance that user and quality requirements are obtained, agreed upon, and recorded, they offer no guarantee that these will drive development and get met. Furthermore, they offer no guarantee that any
subsequent changes to these requirements can be taken into account. As it is frequently cited that the only certainty in systems and software development is that of constant requirements change [Brooks 1982, Lehman & Belady 1985], unless such improvements are coupled with techniques which enable these requirements to be considered and reconsidered throughout the entire process, systems and software will still be delivered which fail to realise them.

Recognition of this situation has led to an influx of quality-oriented approaches to development. These basically involve: (a) specifying the user and quality requirements for the proposed system or software; and then (b) using these requirements as a reference point from which to drive, control, and evaluate the development process. Note that the formality of the RS produced in step (a) determines the extent to which step (b) can be automated. Such approaches set about the task of quality assurance through the use of either:

1. **Development methods**, like structured and top-down decomposition [Yeh & Ng 1990].

2. **Dedicated techniques**, like Quality-Function Deployment (QFD) [Brown 1991].

3. **Development paradigms**, like the Quality Improvement Paradigm (QIP) [Oivo & Basili 1992].

4. Separate **quality assurance groups** or **quality engineers**, as exemplified in the software factory concept [Fisher 1991].

Common to such approaches is their dependence upon the ability to establish and maintain some sort of connection between the information that has been elicited as needs, the requirements which have been derived from this elicited information, and the ensuing artifacts in which these requirements have been distributed. In other words, the ability to meet evolving user and quality requirements depends on them being traceable from their origin and throughout their project life.

In summary, RT is important because it is the primary mechanism which allows requirements to drive the development process, the primary mechanism through which any issues relating to requirements change can be handled, and the primary mechanism through which to verify that these requirements are actually met by the resulting system or software. For these reasons, we believe that systems and software quality is directly influenced by the techniques and tools used for RT.

### 3.4. Evidence of an RT Problem

RT is now recognised as a concern in an increasing number of standards and guidelines for systems and software engineering [Dorfman & Thayer 1990]. It is even specified as a mandatory activity in standards such as the DOD-STD-2167A [DoD 1988a]. Some form of RT is also
essential for realising level two of the Software Engineering Institute's Capability Maturity Model for software [Humphrey 1988] and for realising Clause 4.8 of the International Organization for Standardization's software development and maintenance standard ISO 9001 [BSI 1987]. This concern has further been reflected by a growing research interest in the area [IEE 1991, IEEE 1995, Pohl et al. 1994, Thayer & Dorfman 1990], along with the introduction and enhancement of dedicated in-house and commercial tools for providing RT. We return to tool support for RT later.

Despite the many advances, RT remains cited as a key problem area confronting industry. One need only look to the variety of software and system disasters to recognise how improved RT could alleviate many of the issues cited as responsible for these disasters [Wiener 1993]. Evidence of an RT problem has also been prominent in a number of industrial reports. These draw attention to the lack of substantive guidance as to how RT is best achieved. This means that, for each project in which RT is required, it is difficult to determine and agree details like: (a) what data needs to be traceable; (b) what linkages need to be made; and (c) how, when, and by whom does all this need to be established and maintained. Furthermore, it is difficult to map the chosen scheme on to the available tool support and on to the established organisational arrangements. The amount of time and effort that can be required to achieve a workable RT scheme is all too often underestimated. Although the academic literature also frequently cites RT as a problem area, it rarely enters into any further detail. However, most evidence of an RT problem can be found informally, either through conversations with practitioners involved in systems and software development or through electronic news groups and bulletin boards.

4. State-of-the-Art in Requirements Traceability

In this section, we review the various ways in which RT is currently practiced. We first outline the sources of the review and our organisation of the material. We then describe the basic techniques commonly used to achieve RT and explain how these are incorporated in various approaches to development. Finally, we describe the different ways in which automated support has been provided to operationalise these techniques and approaches.

4.1. Review Sources and Organisation

A surprising number and variety of tools can be construed as offering some form of support for RT, be this in the form of version control, configuration management, or database management. For this reason, we reviewed over one hundred commercial tools and research products to analyse the type and quality of support they offer for RT, informed by accompanying documentation and marketing brochures. This review was supplemented by discussions with practitioners using some of the tools, conversations and practical demonstrations with tool vendors at trade fairs, and some limited hands-on experience where possible. We organise our material using the following classification scheme:
Basic techniques, which are the explicit mechanisms through which RT can be achieved.

(2) Approaches, which incorporate the basic techniques, and so achieve RT as a by-product of their use.

(3) Automated support, which serves to operationalise both (1) and (2) above.

4.2. Techniques

We identified a number of basic techniques through which RT can be actively achieved. These ranged from variations on cross referencing schemes, through document-centred techniques, to more elaborate structure-centred techniques. These various types of technique differ in the quantity and diversity of information they can trace between, in the number of interconnections they can represent between information, and in the extent to which they can adapt to reflect changes and so maintain RT throughout a project's life.

4.2.1. Cross Reference-Centred

By simple cross referencing schemes, we mean those schemes which are embedded in the main project documentation itself, as would be the case with phrases like "see Section x" or the repetition of terms and their explanation in an accompanying glossary. These cross references could be automatically supported if the documentation was held in an on-line form by hypertextually linking the two ends of a cross reference or the consecutive occurrences of a term. Examples of those cross referencing schemes which achieve RT include those based on some form of explicit requirements tagging, numbering, or indexing [Evans 1989], as well as those based on expressing and maintaining specified relationships between key phrases [Jackson 1991]. By more comprehensive cross referencing schemes, we mean those schemes which supplement the main project documentation, typically with the addition of specialised tables or matrices, to specifically keep track of cross references and so enable RT. Examples here include the widely used requirements traceability matrices [Davis 1990], as well as their extension into matrix sequences [Brown 1991].

4.2.2. Document-Centred

By document-centred techniques, we mean those schemes which dictate either all or part of the structure and content of the project documentation so as to ensure RT. Examples include those schemes which specify particular forms to fill in, like the templates used in Software Through Pictures (StP) [IDE 1991], as well as the use of hypertextual document templates within the Document Integration Facility (DIF) [Garg & Scacchi 1989]. Examples also include those schemes which use the notion of integration documents, or transformation documents, to store the links between documents created in different phases of development to ensure RT [Borstler &

4.2.3. Structure-Centred

By structure-centred techniques, we mean those schemes which enhance the project documentation to achieve RT, say by restructuring it in terms of an underlying network or graph. These schemes are particularly evident where the focus of RT is the update and propagation of requirements changes. Examples here include those schemes based on the use of assumption-based truth maintenance networks and chaining mechanisms [Smithers et al. 1991], as well as those schemes based on the use of constraint networks and propagation [Bowen et al. 1990].

4.3. Approaches

Some form of RT can often result as a by-product of systems and software development practice. This is because the above techniques are commonly embedded in various approaches to development. We shall therefore provide examples as to how RT can be achieved by those approaches based on the use of models, methods, and languages. Note that those approaches which combine the use of several methods, models, or languages, can also add an additional level of complexity as to how overall RT is achieved and subsequently handled.

4.3.1. Models

By models, we refer to representations of the development process. Examples of how RT can be provided by the use of models include:

(1) Process entity-relationship models, where RT results from the maintenance of relationships between entities [Hamilton & Beeby 1991].

(2) The DesignNet model, where RT results from the combined use of and/or graphs and Petri nets [Liu & Horowitz 1989].

(3) The SODOS Software Documentation Support model, where RT results from its documentation strategy [Horwitz & Williamson 1986a, Horwitz & Williamson 1986b].

(4) The REMAP model for the representation and maintenance of process knowledge, where RT comes from the primitive concepts and relationships of its conceptual model used for RE [Ramesh & Dhar 1992].

4.3.2. Methods

By methods, we refer to orderly arrangements of procedures to be followed in the development process, accompanied by guidelines for their execution. Examples of how RT can be provided by
the use of methods include:

(1) The Planning and Design Methodology (PDM), where RT results from cross referencing in the series of steps that are followed to transform initial requirements into a structured specification document [Mays et al. 1985].

(2) The Requirements Specification and Traceability Methodology (RESPECT), where RT results from simple on-line cross referencing and from more comprehensive RT matrices [Edwards & Howell 1991].

(3) Quality Function Deployment (QFD), where RT results from matrices being combined end-to-end to form anything from "houses" to "cities" of quality [West 1991, Wolfe 1994].

In addition, RT is frequently inherent in the structuring procedures of formal methods [Cooke & Stone 1991], object-oriented methods [Henderson-Sellers & Edwards 1990], and goal-based methods [Dardenne et al. 1993, van Lamsweerde et al. 1990, van Lamsweerde et al. 1995].

4.3.2. Languages

By languages, we refer to notations used to describe the development process, along with rules for their use. Different degrees of RT are often inherent in the basis of particular requirements languages used in development, such as:

(1) The Requirements Statement Language (RSL) [Bell et al. 1977, Davis & Vick 1977], which contains elements like TRACES_FROM and TRACES_TO in the features of its language specifically to ensure forwards and backwards RT.

(2) The Requirements Modelling Language (RML) [Greenspan 1984, Greenspan et al. 1994], where forms of RT can be achieved as a consequence of its structuring and organising principles used to build requirements models expressed in RML, since binary semantic relations are maintained between the objects used.

(3) The Agent-Oriented Language for Building and Eliciting Requirements for Real-Time Systems (ALBERT) [Dubois 1994], since the language's primitive constructs and structuring mechanisms were specifically designed to enable forms of RT, and further because mathematical relationships can be expressed between successive information states to represent various types of sequencing between them.

4.4. Automated Tools

Paper-based implementations of the above techniques and approaches can soon become too unwieldy and unmanageable to maintain RT. As a result, these are increasingly embodied in
various types of computer support. As it is not possible to review all the commercial tools and research products which support RT in some way here, we consider how specific classes of tool achieve RT. In Section 5, we go on to provide a more detailed analysis of the RT support that each of these offers.

4.4.1. General-Purpose Tools

By general-purpose tools, we refer to those tools which can be configured to support numerous activities. Examples include hypertext editors, word processors, spreadsheets, database management systems, and prototyping tools. Such tools can be hand-configured to allow previously manual and paper-based RT tasks to be carried out on-line. This generally involves establishing cross references between project documents, or document fragments, and placing conditions upon their automatic update. Illustrations as to how RT can be achieved using hypertext technology, or alternatively by combining a spreadsheet front-end with a back-end database, can be found in [Kaindl 1993] and [Watkins & Neal 1994] respectively.

4.4.2. Special-Purpose Tools

By special-purpose tools, we refer to those tools which have been pre-configured to support single and well-defined activities. In the context of this thesis, we are primarily concerned with those tools which offer support for individual activities carried out in RE, of which some can achieve restricted types of RT through use of the tool. For example:

(1) The KJ-editor assists with the organisation of evolving ideas and, in so doing, provides traceability between the vague ideas and the more concrete requirements that come out of this process [Takeda et al. 1993].

(2) PORC is an in-house tool which assists with the extraction of requirements from interview transcripts and, in so doing, provides traceability between the interview transcripts and the requirements derived from them [Langford 1991].

(3) The T tool assists with test case generation and, in so doing, provides traceability between requirements and their test cases [Sodhi 1991].

4.4.3. Workbenches

By workbenches, we refer to where a collection of the above types of tool are organised to support a coherent set of development activities, thus giving rise to computer-aided software engineering (CASE). The support provided for RT depends on the focal activity of the composite tool. Although many types of tool fall into this class, we are mainly concerned with those upper-CASE workbenches which provide support for RE-related work within their scope. This is in contrast to
lower-CASE workbenches which emphasise construction, testing, and implementation work. For clarity, we distinguish the two particular types of upper-CASE workbench we are interested in here as either conventional workbenches or dedicated RT workbenches.

Conventional workbenches generally support the activities of requirements definition, analysis, and design, and they frequently provide some degree of support for RT. Even if a baselined RS is taken as a starting point from which to drive the analysis and design activities, coarse-grained RT is commonly provided between the requirements in this RS and their realisation in the following phases. In conventional workbenches, RT support can be explicit and through the use of specific RT components, like through a requirements tracking matrix in CONQUEST [SACO 1994], and through a Coupling Module in AGE [Keys 1991]. It can otherwise be implicit and from adherence to the advocated approach to RE, like from following the requirements-driven design approach in RDD-100 [ALC 1994]), else as a by-product of having carried out other activities using the workbench tools, as evident in the Requirements Apprentice [Reubenstein & Waters 1991].

Dedicated RT workbenches specifically support RT and requirements management. In such workbenches, all the tools and activities are configured to ensure RT throughout the RE process. Typically centred around a database management system of some form, they comprise dedicated tools for documenting, parsing, editing, decomposing, grouping, linking, organising, partitioning, and managing requirements. They further provide varying degrees of integration potential to help ensure RT throughout the later development phases when third-party tools are used to assist with those activities outside their scope. Examples of dedicated RT workbenches include:

1. The Automated Requirements Traceability System (ARTS) [Dorfman & Flynn 1984, Flynn & Dorfman 1990].
2. The Dynamic Object-Oriented Requirements System (DOORS) [QSS 1994].
3. RTrace [SMI 1990].
5. The Requirements and Traceability Management System (RTM) [MST 1993].

4.4.4. Environments and Beyond

By environments and beyond, we refer to those tools which provide support for the entire systems and software lifecycle, perhaps by integrating those workbenches which support analysis, design, coding, testing, and management activities into a unifying environment in which data can be shared. Here, although RT can potentially be provided throughout a project's life, it is the basis used for this internal integration which tends to define how RT is established and how well it can
Part I: Introduction

be maintained. This can either be through the use of:

(1) A **common language**, such as the Input/Output Requirements Language (IORL) in the Technology for the Automated Generation of Systems (TAGS) [Sodhi 1991].

(2) A **common structure**, such as the relations of an Entity-Relation-Attribute Model in Genesis [Ramamoorthy et al. 1988].

(3) A **common method**, such as the Information Engineering Method (IEM) in the Information Engineering Facility (IEF), where horizontal RT is inherent in its technique integration and vertical RT is inherent in its model transformations [Hares 1992, TI 1988].

(4) A **sophisticated repository structure**, and/or a **specialised RT tool**, in those environments where a number of interlocking tools have been combined to support many languages, methods, or structures. For example, through the object-oriented database of the System Level Automation Tool for Engineers (SLATE) [TD 1995], or through the RqT component of Teamwork which uses RT matrices to relate requirements and deliverables [CT 1992].

Of additional interest are those open environments with the flexibility to incorporate third-party tools. These increasingly provide RT support through the use of powerful repositories and underlying database management systems, used to relate the products of the individual components, as exemplified by the Digital CASE Environment [Sodhi 1991]. Also of interest are those environments which embed an underpinning meta-model of the development process to enable customised support. An example here is Cradle, where support for RT is driven by its innovative and extensive approach to cross referencing [SSSL 1995a, SSSL 1995b].

5. Critique of the State-of-the-Art

In this section, we critically analyse the state-of-the-art in RT, by focusing on the different ways in which the techniques and approaches to RT are generally supported by tools. Although there are many heuristics and guidelines upon which one could base the evaluation of these tools, such as [Anderson 1989, QED 1989, Sodhi 1991], we have based our critique on criteria suggested by practitioners in the empirical studies we conducted. These include:

(1) What priority does the tool give to RT?

(2) What mechanical and analytical support is provided by the tool for establishing RT?

(3) Who has to establish the RT when using the tool?

(4) What kinds of requirements-related information can be made traceable by the tool?
(5) What is the breadth and longevity of the RT provided by the tool?

(6) What are the main tasks and job roles which are supported by the RT the tool provides?

Based on a brief account as to how general-purpose tools, special-purpose tools, workbenches, and environments each deal with such questions, we then list what we perceive to be their main strengths and weaknesses with respect to RT support. We finish by summarising where we believe the current state of RT practice lies.

5.1. Priority Given to RT

Although any general-purpose tool can potentially be configured for RT purposes, RT is not a priority of the basic tool. For this reason, it is not unusual for the focus of activities to become tool configuration and maintenance, as opposed to RT itself. The priority assigned to achieving RT in any hand-crafted application is wholly dependent on the application developers.

Special-purpose tools which support single requirements-related activities, say a specific requirements analysis technique, often provide for some form of RT. However, as RT is typically achieved as a by-product of ordinary tool use to support the activity in question, RT is generally not the tool's main priority.

Amongst workbenches, the priority given to RT varies dependent on the focal set of activities for which the internal tools have been configured. Where these activities are RT and requirements management, as in the dedicated RT workbenches, RT is the main priority. In those workbenches where the focus is more widespread, to support the requirements, analysis, and design phases, RT is often one of many side concerns, so generally provided as a by-product of use.

With environments, as well as with the conventional workbenches, RT is typically regarded as a side concern. Its priority depends on the types of tool contained in the environment and how such tools have been internally integrated. Amongst environments, the highest priority for RT can usually be found in those which contain dedicated tools or toolsets for RT purposes.

5.2. Support Provided for RT

No explicit support is provided for RT in general-purpose tools. As provision for RT must be hand-crafted, the resulting guidance and assistance that is offered tends to depend on the effort expended in putting in place either manual or automated procedures for establishing and maintaining RT. The sophistication of the underlying RT scheme which is being provided with computer support is a major factor which dictates the degree to which RT will be supported.

With special-purpose tools, support for RT is mainly implicit in the framework provided for
carrying out the main activity of the tool. This means that the mundane and repetitive tasks necessary to provide basic RT are typically automated as a result of proper tool use. Explicit guidance and assistance for RT itself is not really apparent.

In conventional workbenches, the support for RT reflects that offered by the special-purpose tools, namely the automation of some mundane and repetitive tasks needed to achieve RT. In dedicated RT workbenches, support is more explicit, as RT is now something which is consciously achieved. These generally offer some form of guidance and assistance for RT. Guidance for RT often comes through adherence to the RE approach and the work steps supported, which are typically top-down and decompositional, and by a priori definition of the information to collect and the link types to establish. Assistance for RT often takes the form of parsing textual documents to identify requirements, tagging these requirements, then establishing syntactic links between them. Further assistance is generally offered through their repositories which manage the simple bookkeeping tasks and enable rudimentary forms of requirements checking. However, it is rare that any real analytical support is provided for RT in any of the workbenches.

Amongst environments, RT is mainly provided as a by-product of other development activities, so through coordinated tool use and adherence to the development philosophy supported. The extent of support therefore depends on this philosophy, the internal integration strategy for its tools, and/or its repository structure. More guidance and assistance tends to be on offer if dedicated RT tools or toolsets are included in the environment, or if RT is an explicit concern of the development philosophy it embeds. On-going support for RT is further possible where the repository can manage large amounts of information, versions, variants, and can reconfigure itself in the light of any requirements changes.

5.3. Who Does the RT

With general-purpose tools, RT is not generally achieved as a team effort, but rather as a result of individualistic working. To avoid disintegration, they are best used by a single user who records requirements activities after they have happened, and then establishes RT through them. This is because, unless firmly grounded in an organisationally accepted RT scheme, they tend to provide no common or consistent framework for RT. They often encourage immediate and ad hoc solutions.

Most special-purpose tools support individualistic working, so the RT provided is typically a consequence of individual working in the development process. This individual need not be familiar with RT issues themselves. Those special-purpose tools which support group activities, like the brainstorming of requirements amongst a team, make both the group process and its results traceable. So in such cases, RT can be said to have been achieved by the group involved, though individual members do not need to be aware of the RT they are making possible.
Dedicated RT workbenches tend to be used as after-the-event documentation tools, either by an individual, or by a small group of individuals. This is because they can be difficult to adapt to the immediate working practices of requirements engineers. Although these workbenches increasingly provide support for multiple users, through the use of repository partitioning, locking mechanisms, and RT-directed approaches to requirements development, the resulting RT is still generally more successful when the RT is coordinated by single users. This is because RT is often difficult to maintain when there is much concurrent work. The potential richness of the requirements information and the RT through this can easily be lost. Even with the most dedicated forms of RT support, it is essential that all the stakeholders have bought into the underlying RT scheme it supports.

With environments, multiple users are commonly supported through shareable repositories and groupware-related techniques, a situation again similar to that for the more conventional workbenches. The coordination and integration of separate activities is common and can be supported quite extensively through emerging concepts such as workspaces and views. However, the ability for multiple users to be successful in achieving and maintaining RT depends upon an agreed baseline RS, and often a rather static one, as well as upon strict project partitioning policies. RT can therefore deteriorate when this baseline is not stable and when overall control of the project or coordination for RT is lacking.

5.4. Requirements-Related Information Made Traceable

General-purpose tools offer the flexibility to trace any information which can be input to the tool, be this textual, graphical, and so forth. They just need to be configured to do so. This is increasingly the case now that many tools, such as word processors, diagram editors, and audio recorders, can integrate their data as compound documents. In this way, they are potentially able to trace any requirements-related information requested, so long as it can be captured in some tangible form and digitised.

In contrast, special-purpose tools tend to predefine both the amount and type of information that can be input and made traceable. This is typically restricted to that information which is necessary to carry out the specific activity the tool supports. Under such conditions, only a limited amount of requirements-related information can ever be traced, and this is rarely adaptable.

Those workbenches supporting a broad spectrum of requirements activities offer the potential to trace a wider range of requirements-related information, which often includes information in multiple media. However, most seem to impose some arbitrary limit on the amount and type of information that can be traced. Although dedicated RT workbenches tend to make it possible to trace a greater wealth of information about requirements, to date this has predominantly been textual, with graphics only recently beginning to be supported. Where additional information like
rationale can be recorded as informal notes, these do not regularly integrate with the mainstream information, so are generally of limited use for RT purposes.

Environments bring with them the potential to trace all the project information related to a requirement if all of a project's activities can be carried out within the environment's boundaries. However, as their tendency is to focus on supporting down-stream activities, RT is generally only maintained from the baselined RS. The traceability of requirements versions and variants can vary widely in quality dependent on the structure of underlying repositories and also on their version control and configuration management mechanisms. In addition, the traceable requirements information also tends to be thinly spread, so is not particularly detailed.

5.5. Breadth and Longevity of RT Support

General-purpose tools are typically configured to address the immediate activities and needs for which RT is required. They are unlikely to support all aspects of a project and are usually not well integrated with specialised development tools. RT can therefore degrade with large amounts of information, users, and time, unless explicitly prepared for. For the same reason, they can be poor at handling requirements changes and evolution. The breadth and longevity of the RT support firmly depends on the effort and foresight of those configuring general-purpose tools for RT purposes.

As special-purpose tools support a specific requirements-related activity, they only provide RT for a snapshot in time, as they neglect the on-going management of any RT they put in place. The breadth of RT support depends on horizontal integration with other requirements-related tools. The longevity of RT support depends on vertical integration with tools supporting the previous and subsequent phases of a requirement's life.

RT is provided throughout the activities and development phases supported by specific workbenches. As most workbenches are predominantly forwards-engineering tools, RT can deteriorate with progression to later phases, as it can be difficult to reflect later external work in this earlier work. For this reason, backwards RT is generally their weak point, which in turn means that any iteration is often not adequately accounted for. Although dedicated RT workbenches provide the potential for a rich breadth of RT support throughout the requirements phases, the longevity of this support depends on vertical integration with other lifecycle tools. Interestingly, vertical integration to exploratory tools in which the input to these workbenches are determined appears to be more or less lacking, which in turn means that the RT support is unnecessarily limited for the very early phases.

Although the breadth of RT support provided by environments is rarely as rich as the dedicated RT workbenches, they do make it possible to provide RT throughout a project's life. However, this
does tend to start from a relatively static RS baseline, since design activities and automatic code
generation almost always take priority. The tightness and granularity of this RT, which is an
important factor in the ability to reflect later changes to requirements in the baseline, again
depends on the underlying repository and the degree of internal integration amongst the tools. RT
can therefore deteriorate over time, due to iteration problems, and due to poor feedback regarding
any deviations that have been taken.

5.6. Tasks and Job Roles Supported by RT

General-purpose tools can be tailored so that the RT they provide can be used to support any task
and job role in systems and software development. What is problematic though, is the ability to
configure these tools to meet different needs simultaneously, which would require some in-depth
study prior to application development. This would counteract many of the advantages for using
general-purpose tools in the first place.

With special-purpose tools, RT is provided to specifically assist the activity the tool supports, and
so to support the predefined developmental role of the tool user. Their task-specific frameworks
constrain the domain of working and are difficult to configure to meet other tasks or to support
other roles in systems and software development.

The RT provided by workbenches can support a breadth of activities within the concern of the
tool's domain. They can provide dedicated support for specific requirements and management-
related tasks, such as assistance with requirements checking, monitoring, and reporting. They can
support the activities of those transforming requirements into design. Furthermore, they are often
configurable to support some tasks that occur in additional project phases, and are frequently used
by those involved in coding or maintenance. One concern is that the RT provided by dedicated RT
workbenches tends to support managerial activities, and rationalised RE activities, rather than the
actual activities of those directly involved in producing and using the requirements.

The RT provided by environments can assist a wide range of lifecycle-wide tasks and roles. For
instance: (a) those related to coding, say examining how a set of requirements can be realised in a
specific module; (b) those related to maintenance, such as change impact analysis; and (c) those
related to management, such as progress reporting. However, not all are equally well supported,
requirements management can be particularly weak, and most support is inclined towards those
activities related to requirements use rather than their production and evolutionary refinement.

5.7. Main Strengths

Based on the above accounts, we list what we consider to be the main strengths of the various
forms of automated tool support for RT. This list is first summarised in Table 1.
Part I: Introduction

<table>
<thead>
<tr>
<th>General-purpose</th>
<th>Special-purpose</th>
<th>Workbenches</th>
<th>Environments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Offer the ultimate flexibility for RT.</td>
<td>Provides tight RT to support a specific activity.</td>
<td>Provides fine-grained forwards, backwards, horizontal and vertical RT, especially in the RE phases.</td>
<td>Provides on-going RT throughout a project and across versions and configurations of requirements.</td>
</tr>
<tr>
<td>Readily available in most organisations for RT use.</td>
<td>RT is usually a by-product of use.</td>
<td>Added value is often provided as a consequence of RT.</td>
<td>RT is often possible despite distributed work settings.</td>
</tr>
<tr>
<td>Usually sufficient for the RT needs of small projects.</td>
<td>RT of focused group activities is often possible.</td>
<td>RT is possible across requirements artifacts in multiple media.</td>
<td>Can offer a degree of freedom in the choice of RT support.</td>
</tr>
</tbody>
</table>

Table 1: The main strengths of the different classes of tool support for RT.

5.7.1. General-Purpose Tools

(1) They offer the ultimate flexibility to provide customised and comprehensive RT to suit individual projects and organisational needs.

(2) They tend to be readily available in organisations, so if a simple and agreed RT scheme is in place, they can offer the most cost-effective solution.

(3) They are generally sufficient for the RT required in small and short-term projects, especially those with few participants, as these participants can informally deal with the control aspects of maintaining RT.

5.7.2. Special-Purpose Tools

(1) They can provide tight RT which is sufficient for the immediate needs of particular requirements-related activities.

(2) They do not require their users to concentrate on achieving and maintaining RT, like routinely specifying and updating any traceability links, so they have less scope for manual error in their RT.

(3) Those tools supporting group activities often provide some traceability of these activities, as well as of their intermediate and end products, so is particularly useful if these are requirements exploration activities.

5.7.3. Workbenches

(1) Dedicated RT workbenches provide adequate RT from and back to the information which is
initially input to the tool and through a potentially rich breadth of requirements-related activities which they often support. In particular, they can offer fine-grained horizontal and vertical RT within the RE phases, both forwards and backwards.

(2) Dedicated RT workbenches offer additional benefits as a consequence of their RT support, ranging from facilities for testing requirements and their allocation, through to facilities for generating various RT reports.

(3) Conventional workbenches, and increasingly the dedicated RT workbenches, offer the ability to trace requirements between the various types of multiple media in which they may be represented.

5.7.4. Environments and Beyond

(1) They have the potential to provide on-going RT throughout a project's life, so offer the most depth of coverage, or vertical RT. With the support they generally provide for version control and configuration management, horizontal RT can also be well supported.

(2) Those environments with sophisticated multi-user facilities have the most potential to achieve RT amongst distributed group working situations.

(3) Those environments which are open to integration with third-party tools provide the most flexibility in the choice as to how RE is carried out, so also as to how the RT is supported.

5.8. Main Weaknesses

Also based on the above accounts, we list what we consider to be the main weaknesses of the various forms of automated tool support for RT. This list is first summarised in Table 2.

<table>
<thead>
<tr>
<th>General-purpose</th>
<th>Special-purpose</th>
<th>Workbenches</th>
<th>Environments</th>
</tr>
</thead>
<tbody>
<tr>
<td>RT can have a high start-up cost.</td>
<td>The RT provided is generally rigid.</td>
<td>RT can become the focus rather than RE.</td>
<td>Often only provides coarse-grained RT.</td>
</tr>
<tr>
<td>Providing for and maintaining RT can require much effort.</td>
<td>The RT provided generally has a limited life and use.</td>
<td>RT depends on stakeholder buy-in.</td>
<td>Backwards RT is generally poorer than forwards RT.</td>
</tr>
<tr>
<td>The RT provided can be unpredictable.</td>
<td>The RT provided is frequently not even noticeable.</td>
<td>RT beyond the RE phases can vary widely.</td>
<td>RT can be difficult to maintain across tool components.</td>
</tr>
</tbody>
</table>

*Table 2: The main weaknesses of the different classes of tool support for RT.*
5.8.1. General-Purpose Tools

(1) An extensive RE exercise can itself be required to understand the RT which needs to be supported. What often results using such tools is often little more than an electronic version of paper-based RT which does not scale up to industrial strength.

(2) Not only can they require much work to initially configure, establishing and maintaining RT within their frameworks can involve many mundane, repetitive, and error-prone activities. Any subsequent tailoring can be problematic unless this ability has been explicitly prepared for.

(3) They generally exhibit poor control and integration potential, so provide no guarantee as to the usefulness, usability, and longevity of the RT provided.

5.8.2. Special-Purpose Tools

(1) They only provide restricted forms of RT, between limited types and amounts of requirements-related information, so this has limited life and use.

(2) They typically exhibit poor integration and information management potential, thus preventing fuller and longer RT support.

(3) The RT provided is built-in, not negotiable, and often not even visible.

5.8.3. Workbenches

(1) Although dedicated RT workbenches attempt to be holistic, none support all the possible requirements-related activities that could be required in a project. Typically, a top-down approach to requirements development is enforced, classification schemes are predefined, and a relatively static baseline RS or source documents are pre-empted. Support for the production and iterative enhancement of these starting points is largely absent. As the RT depends on their correct use, the main concern can become RT, rather than engineering the requirements.

(2) Much manual intervention can be required to maintain RT if there is need to work around what the dedicated RT workbench dictates. Its success depends on buy-in from all the stakeholders, as the slightest deviation can lead to the failure of the whole RT scheme, hence why they are often used as after-the-event documentation tools and for their managerial facilities.

(3) Workbenches tend to integrate only with a few pre-specified tools, making it difficult to support the RT of any early work done to define the problem space, and therefore to provide
on-going RT with later requirements changes in subsequent phases.

5.8.4. Environments and Beyond

(1) RT is typically coarse-grained and dependent on following a step-wise approach to development. Their RT is generally comparable with that of conventional workbenches, often substituting breadth of RT support with length, though not as rich and flexible as that provided by the dedicated RT workbenches.

(2) The tightness of the RT varies, so any iteration or later requirements changes can sabotage the potential for on-going RT, frequently a consequence of poor backwards RT. This is because they often cannot account for any manual intervention that occurred in forwards engineering, a common occurrence in those environments which have automatic code generation.

(3) The increasing flexibility for RE which comes with those tools that are open to external integration is typically counterbalanced by poorer RT. This is because individual expertise is generally acquired with different tools and the success of the overall RT is typically dependent on continuity.

5.9. Summarising the State-of-the-Art

It has been noted that the majority of the development tools on the market do not cover RT [Polack 1990]. More specifically, that even fewer of these tools provide support for the particular RT requirements now enforced by the DOD STD-2167A [DoD 1988a]. We have shown that those which do address RT use the same basic techniques, so differ only cosmetically. They also differ in the amount of time, effort, and manual intervention they require to establish RT, and then to maintain and make use of the RT they provide. The type and extent of support depends on the underlying assumptions tools embed about RT and on the particular problems and concerns that they focus on.

When we consider the dedicated RT workbenches, it is interesting to note that today's variants are not that different from their early predecessors, as can be seen through a simple comparison between ARTS and RTM. Although many of the technical issues to do with RT are now well addressed, little useful guidance is generally provided as to the requirements information to trace between in particular situations, and as to the traceability relations that are needed to provide different forms of RT between this information. The same applies to guidance concerning how these tools can be adapted to unique working practices without losing their RT potential. It is therefore not surprising to observe that these tools are not as widely used in practice as the importance of RT would suggest and that RT problems are still cited where they are used. These
limitations have been reflected by a preference for using general-purpose tools in practice [Lubars et al. 1993]. No doubt this is because they can most easily be individually configured to support in-house RT problems and needs.

6. Directions in Requirements Traceability

In this section, we point to the most recent and popular commercial RT tools, to consider why they lead the market in dedicated RT support. In so doing, we provide some insight into the status of commercial RT support. Although we do not review those environments which provide an end-to-end solution to RT here, we believe that these are likely to prove more popular in the near future. Following this review, we then indicate the focus of contemporary research related to RT, paying particular attention to formal transformations of requirements and RT modelling.

6.1. Popular Commercial Tools

At the present time, three commercial tools dominate the market for RT support. These are:


(3) **RDD-100**, the Requirements-Driven Design system, developed by Ascent Logic Corporation, U.S.A. [ALC 1994, Alford 1993].

Available on most common platforms, each tool has well over one hundred customer sites, though the numbers for each continues to increase. Of these, we would classify DOORS and RTM as dedicated RT workbenches, and classify RDD-100 as a more conventional upper-CASE workbench dedicated to requirements analysis, modelling, and design. Here, we provide a brief overview of each tool. We describe the approach they support to requirements development, the RT they make possible, and the factors which contribute to their popularity. It must be noted that these tools are each in the process of continuous and rapid evolution. Any review is therefore likely to become outdated fairly quickly.

6.1.1. **DOORS**

DOORS is a multi-user tool for organising and managing information. It focuses on the management of requirements and their traceability. It advocates a structured approach to developing requirements documents. It provides document templates for following many of the
popular software engineering standards, such as the DOD STD-2167A, DOD STD-499B, and ESA PSS-05, and provides the ability to construct new templates. Object-oriented techniques are further used to build up hierarchical sets of requirements and to handle the complexity that comes from having large amounts of information and interlinkages.

Requirements can be directly imported to DOORS from existing text files. The requirements in these source documents are identified and extracted, either manually or through the use of a programmable requirements parser, though all inputs are treated as black boxes that cannot be edited or reviewed from within the tool. The extracted requirements text is then structured and marked with predefined attributes, like a unique numerical identifier for each requirement, and then an unlimited type and number of user-definable attributes can be attached to each requirement to define aspects such as their priority, status, rationale, and so forth. Each requirement further has its own change revision history, which is a textual list incorporating details like who made changes, when, and details of the requirement both before and after the change. RT is primarily carried out using the attributes. These attributes can be entered textually with each individual requirement, else graphically using cross reference tables between two documents, or by manipulating back-to-back tree views of documents. Due to this use of attributes, the types of RT supported are extensible and user definable, and requirements can be traced between numerous documents stored within the tool.

DOORS is popular for a number of reasons, including:

1. The ability to make arbitrary traceability links between information sets. The ensuing ability to construct hierarchies of heterogeneous types of document means it handles the RT in large projects by decomposing their documents into lots of smaller ones and managing the interactions between them.

2. The ability to integrate with a number of third-party tools to support other development activities. In addition, through the provision of an open tool interface builder, it can further be configured to integrate with customer's own in-house tools.

3. The provision of a scripting language, the DOORS Extension Language (DXL), which enables the functionality of the tool to be extended and customised. Libraries of useful RT-related functions, such as the costing of requirements changes, can be developed.

4. It does not require much expertise and lengthy training to use the basic features of DOORS, though increased utility does come from learning to use DXL.

6.1.2. RTM

RTM is a multi-user tool which focuses solely on RT and requirements management. It is built on
Part I: Introduction

a generic model of the development process as it has been observed in practice over many years. It is probably the most popular dedicated RT tool amongst practitioners at the moment and shares much in common with what we have said about DOORS, including a relatively short learning curve to be of use to both skilled and unskilled engineers.

RTM contains a number of internal and coordinated tools which serve to capture, control, and maintain requirements and their interrelationships. Source documents are input to the tool, requirements are extracted, either manually or automatically, and then keywords, rules, attribute values, and relationships are defined for these extracted requirements. These requirements can then be further expanded or focused using a suite of editors. Through this process a particular instance of an RTM requirements model is constructed. Note that for each project, users can specify the RT scheme to be supported using a graphical schema definition. This specifies the artifacts and objects that will play a role in the project, along with attributes to be associated with them, as well as the relationships that can exist between these artifacts. The RT it provides therefore follows from a top-down decompositional approach to requirements development and the granularity of this RT is entirely dependent on the user's choice of RT scheme.

RTM is popular for a number of reasons, including:

(1) The ability to be pre-configured to address different project RT needs and project lifecycles. This means that it does not impose strict pre-conditions on use, but guides use once the RT scheme to be used has been configured using its graphical schema definition facility.

(2) The ability to interface directly with a number third-party tools to support other development activities, most notably with RDD-100. This again offers the potential for lifecycle-wide RT.

(3) The ability to automatically identify and capture requirements from source documents using an auto stripper tool. This facility was developed because one of the main issues RTM intended to address was the problem of identifying requirements in large document sets. It works by pattern matching and implements textual guidelines concerning what is and what is not a requirement, how much text should be captured, how the context should be captured, what attributes should be assigned to requirements, as well as many other rules of thumb. It is continually being enhanced to take account of advances in relevant areas.

(4) Its database partitioning option manages requirements and their traceability across disconnected networks. As this facility enables both multiple and selective partitions, it is particularly suited to the growing culture of subcontracting in industry.
6.1.3. RDD-100

RDD-100 is a multi-user integrated toolset with built-in support for the Requirements-Driven Design approach to development. Its main focus is on requirements modelling, analysis, and design. As the tool neither provides any text editing facilities, nor integrates with desktop publishing tools, source requirements documents are input in ASCII format. By then following a top-down decompositional approach to requirements development, the represented requirements in these documents become structured, and hierarchical traceability trees get built in the process. The tool further supports the production of a full underlying and executable system behaviour model.

Entity-Relationship-Attribute modelling is the basis through which RT is made possible. Requirements are represented as entities and relations define named bi-directional links to other requirements or information stored within the tool. As there is one underlying model, from which alternative views and simulations can be dynamically produced, this enables the RS to remain a living document. The RT provided is particularly useful for simulation, like for tracing defined requirements to their behaviour, tracing the allocation of requirements to components of the system architecture, and so forth.

RDD-100 appears to be a popular tool amongst those practitioners who are more technically-minded. This is because it consists of a very powerful, but somewhat unwieldy and overwhelming set of capabilities, which give it a steep learning curve. As it complements both DOORS and RTM, with regard to both its functionality and scope, it is perhaps best used in conjunction with either of these. It can currently integrate with RTM and this has been shown to provide quite a comprehensive solution to RT.

RDD-100 is popular for a number of reasons, including:

1. The richness of the requirements-related information it captures. It captures more than just the end product of the process it supports, so is particularly good at providing RT through the requirements development process.

2. The ability to display multiple and various views of an underlying organic model of behaviour and so reflect changes in one view in others. The modelling representations it supports here range from simple data flow diagrams through to functional flow block diagrams.

3. The facilities it provides for requirements simulation, data sharing, dynamic verification, report generation, and more.

4. The fact that it is a tight-fit system. Although this means that it predefines how the tool is to be used, it also means that it is in a much better position than the other two tools to provide
active support for requirements development in numerous ways, thus setting the type of RT it provides apart from the other two.

6.2. Formal Transformations

Automation-based software engineering, or automatic programming, has been an active research topic for many years [Balzer et al. 1983, Balzer 1985]. This aims to automatically transform high-level problem specifications into source code and to provide the ability to replay these transformations following change. This paradigm would bring with it the ability to eliminate many of the problems connected with tracing requirements from and back to an original RS. Although the formal methods, refinement techniques, and tools needed to support this paradigm are becoming readily available, a number of obstacles are still reported [Wing 1990].

Most of the recent work in this area has been examining either:

(1) Ways to proceed from an informal specification of requirements to a form amenable to such automatic transformation [Bubenko et al. 1994, Fraser et al. 1991].

(2) How tools could be developed to implement the transformational algorithms needed [Borstler & Janning 1991].

(3) How aspects such as user requirements could be tied into these formal approaches [Duke & Harrison 1995].

(4) How requirements objects could be structured as module clusters, by applying the principles of hyperprogramming, to further promote the derivation of formal specifications [Goguen 1990a, Goguen 1990b].

(5) The utility of various types of formal transformation to, in turn, provide support for different kinds of RT [Luqi & Goguen 1993].

6.3. RT Models

Recent directions in RT-related research have seen the development of numerous RT models. The objective of such models is to suggest the information that needs to be recorded and the link types that need to be established between these different information types to meet the RT needs of practitioners. They further aim to offer ways to prioritise and organise all this information for RT purposes.

RT modelling is a fundamental aspect of the ubiquitous ESPRIT NATURE project and is demonstrated in a prototype RE environment called PRO-ART [Jarke et al. 1993a, Jarke et al. 1993b, Jarke et al. 1994b, Pohl 1994, Pohl & Jacobs 1994, Pohl et al. 1994a]. Also of interest is:
(a) that work which specifies a rich set of traceability relations to assist a goal-structured
requirements analysis process used to derive requirements specifications [McDermid et al. 1994,
Morris & Coombes 1994]; and (b) work that specifies different types of traceability link that can
exist between requirements and design to support the BORIS integrative method [Curran et al.
1994, Pyle et al. 1993]. The purpose of such work is to investigate useful semantics for
traceability relations in order to give RT more meaning. The next obvious step would be to
formalise these different types of traceability relation to promote more comprehensive forms of RT
analyses. However, the main problem still appears to be identifying an adequate set of traceability
relations for conducting and assisting development in a practical way.

The most significant work in this area comes from the Naval Postgraduate School in Monterey,
California, U.S.A. This work has been driven by the RT needs of large-scale Department of
Defense projects in the U.S. It therefore comprises a comprehensive series of models to achieve
RT between requirements and all the system components created at the various stages of
development in this particular domain [Ramesh & Edwards 1993]. These models include: (a) a
requirements management model; (b) a high end traceability model; (c) a design/implementation
allocation model; (d) a decision making model; and (e) a compliance verification model. As a
consequence of [Gotel & Finkelstein 1993], a further model was added to this series to account for
RT from the project initiation and mission analysis phases of DoD projects [Laubengayer &
Spearman 1994]. Each such model contains a set of permissible information types and a set of
permissible relation types that can exist between this information. For instance, part of such a
model may state that:

\[\text{a <stakeholder> specifies <needs}, \text{ which either:}
\]
\[\text{initiates a <change proposal> that subsequently modifies a <requirement>; or}
\]
\[\text{defines a <requirement>}
\]

Here, the relation types are underlined and the information types are placed in angular brackets.
More details about these models and their use can be found in [Harrington & Rondeau 1993,
Ramesh et al. 1995].

A number of issues arise from this form of RT modelling, like how can a user of these models
distinguish between the sheer number of information types and relationship types when attempting
to categorise? Furthermore, what happens when the project information does not fit neatly into the
model? The DoD requirements management model alone is relatively complex and has sixteen
possible categories of information to account for and twenty three different types of relation. Even
the simplest of schemes for organising and relating information, like the Issue-Based Information
System (IBIS) framework used in the CM/1 tool [CMS 1993], comes up against problems when it
comes to assigning categories in practice. One person's assumption will always be another
person's rationale, as well as a further person's decision. Other important points to consider

62
include: (a) how the different models fit coherently together and stay so; (b) the additional information that could be inferred from the information within such models; and (c) how generic these models are.

7. Persistent Requirements Traceability Problems

In this section, we explain why the current RT support has not fully succeeded in eliminating the RT problem, and also explain why the thrust of current RT research is unlikely to lead to its elimination. In particular, we suggest that the work to date has either been solution-driven or domain-specific. Furthermore, we suggest that this work has not been accompanied by an informed analysis of what the RT problem actually is, so argue an a priori need to define the RT problem. Here, it is interesting to point out that the need to focus more attention on defining problems has recently come to the attention of the wider software engineering community [Jackson 1994, Jackson 1995a].

7.1. Why RT Problems Persist

Despite the ever-growing number of commercial tools which now support RT, their use by practitioners is not that widespread, and RT problems are still cited by those who do use them. Also, despite the burgeoning pockets of academic research, RT remains a widely reported problem area by industry. A few reasons stand out as to why RT problems persist:

(1) The inflated claims for RT that are made by tool vendors are not realised in practice. This is because there are many difficult issues that need to be considered prior to using such RT tools. The problems lie firstly in setting up a shared, consistent, and coherent RT scheme for each project. They then lie in the need for one hundred percent commitment from all the stakeholders and in the need for some overall coordination. With this in mind, many of the problems still being experienced are not technical problems, but human and organisational problems. Although technical solutions are still needed for projects with huge numbers of requirements, like in the U.S. DoD, most of the outstanding problems do not have purely technical solutions.

(2) RT has a high start-up cost and needs continued funding throughout a project. Project funding is often limited at the onset of a project, restricted to those aspects of the project which are tangible and visible, and subsequently allocated in a phase-by-phase manner. This means that short-cuts are often made with RT when there are problems with budget or time. In many projects, RT is not even considered until it is required to start addressing the problems that inevitably arise, by which time it is generally too late. Again, many of the problems here do not necessarily have technical solutions, and these will remain unless RT receives dedicated project resourcing.
(3) There is a situation of double standards. By this, we mean that work on RT does not tend to be an exemplar of recommended RE practice. RT techniques and tools are generally developed and put in use prior to an up-front understanding of what the RT problem is that is being tackled in a particular project or organisation. Indeed, the RE literature tends to focus on briefly documenting the existence and consequences of RT problems, rather than discovering what they really are and what causes them, so is not in a position to examine what is really needed to relieve them.

The first two reasons above suggest the need for changes in organisational and project culture. This is something we can recommend, and even propose guidelines for, but something we can do little more about. In contrast, there is much more scope to do something about the third reason above, not least because an increased understanding is essential for encouraging and enabling progress with the cultural changes that are required.

7.2. A Need for Problem Analysis

We therefore attribute the persistence of RT problems to the lack of any thorough analysis of the RT problem. We need to define what the RT problem actually is if we are to have any chance of identifying any potential technical and organisational solutions to address it. In particular, we need to uncover the issues which underlie current problems, and so identify those which hold the potential for the most long-term and far-reaching improvements. By so doing, we would be in a better position to examine solutions to those we find the most central. An analysis of the RT problem is also essential because we suggest that it falls into the class of what have been referred to as "wicked problems" [Kunz & Rittel 1970 (cited in [Conklin & Begeman 1988])]. A wicked problem is one which can be characterised in three ways: (a) there is no unique solution, but any solutions that are put forward help to highlight what the real problems are; (b) there are multiple stakeholders, though this set of stakeholders is not stable; and (c) there are no ultimate stopping conditions, so only satisficing solutions will ever be found.

8. Summary of Part I

In Part I, we have presented the thesis argument and introduced the topic of RT. We have situated RT within the discipline of systems and software engineering, demonstrated how it is integral to RE, and indicated why RT is crucial in the drive for quality systems and software. We have further explained both how and why there is a problem with current RT practice and also a shortcoming with contemporary research directions. Lastly, we have argued that the relatively slow progress we have seen with eliminating the RT problem is a consequence of a poor understanding surrounding what the so-called "RT problem" actually is. We have therefore suggested that the most appropriate way forward is through a thorough analysis and definition of this problem.
Part II: Problem

1. Overview of Part II

In Part II, we report on a detailed analysis of the RT problem. We first describe the empirical studies that we carried out to collect data about the RT problem, highlight how this was analysed, and then present the main findings. In so doing, we clarify the multifaceted nature of the problem, and also suggest a framework through which to analyse the problem further. This includes a working definition of RT and a distinction between pre-RT and post-RT. We go on to discuss how both types of RT are currently supported and the impact that improved support can have on reducing RT problems. We then argue that inadequate pre-RT is likely to be the cause of RT problems in the longer-term and that improvements here are the most crucial. As a consequence, we delineate those issues which currently compound improvements to pre-RT in practice, and so reveal the challenges involved. Based on these findings, we then specify the key requirements needed to provide and make use of pre-RT, and further examine how they could be addressed. We then show how, despite the numerous advances we point to, a more pervasive problem remains. Identifying those who have been involved in the production and refinement of requirements was found to be the crux of the RT problem which, if suitably addressed, would offer the most leverage for reducing long-term problems. We discuss this focal problem in more detail, describe how it is currently dealt with, and suggest how it could be tackled in a more appropriate manner. Finally, we describe a number of areas which were found to be pertinent to the focal problem, and so helped inform our proposed solution.

2. Empirical Data Gathering

In this section, we provide a description of the empirical data gathering that we carried out to analyse and define the RT problem. We summarise each of the techniques we used and how they fit together. We further outline how the collected data were analysed, to firstly describe the problem, and to secondly determine what is needed to address it.
2.1. Method and Subjects

The shortcomings which result from relying on one source of data in empirical studies are comprehensively reported in both the social studies literature [Moser & Kalton 1971] and in the RE literature [Goguen & Linde 1993]. For this reason, we used a combination of complementary data gathering techniques to collect data about the RT problem. As shown in Figure 2, this was then used to analyse and define the RT problem, determine what is needed to address it, and to further direct the data gathering process itself. We outline each of the techniques we used, explain why and how they were used, what data they provided, and how their data were analysed. For more details about any of the individual techniques, the reader is referred to [Oppenheim 1992, Stewart & Shamdasani 1990], as well as to the previously mentioned sources.
We conducted the bulk of the empirical data gathering with around one hundred practitioners from within a major U.K. company over the period of a year. This was augmented with data gathering from within a much smaller U.K. company. The practitioners from within the smaller company were used for pilot studies and to ratify our findings along the way. To ensure representative coverage within the larger company, the empirical exercises were carried out with practitioners from different functional divisions and based at dispersed geographical locations. Their primary
working areas covered the various aspects of systems and software development, maintenance, and management. Their experience in these areas ranged from nine months to over thirty years. Furthermore, the number, type, and size of the projects they had been involved in also varied widely.

2.2. Surveys and Critiques

We surveyed the literature to gather together many interpretations as to what RT is, why it is needed, and what the current problems with it are perceived to be. Our examination into the state-of-the-art in RT, described in Part I of this thesis, also provided insight into these issues. Together, this information served to assist in the planning and use of the empirical data gathering techniques described below, as well as to supplement the data they obtained. These documentary sources further helped to locate research and development in a number of relevant areas.

2.3. Focus Groups

We carried out five focus group sessions with small groups of practitioners from within the larger company. These were spread across five development sites of the company and involved thirty seven practitioners in total. They were conducted as semi-structured discussion sessions based on a high-level agenda. Each session lasted one hour, was audio taped, and later transcribed in full. A collective summary of the focus group transcripts and a synopsis of the results can be found in Appendix A.1.

Our main objective in conducting focus groups was to find out, in practitioners' own terms, and based on their own experiences: (a) what RT is; (b) whether they experience problems with RT; (c) what their RT problems are; (d) how RT problems materialise and manifest themselves in practice; (e) what further problems RT problems entail and cause; (f) how any RT problems are currently handled in practice; and (g) any suggestions as to where or how any short-term and long-term progress can be made with RT. A further objective was to provide guidance for subsequent data gathering, so to indicate the important questions to follow up in questionnaires, as well as to suggest how such questions could be asked in practitioner terms.

We analysed the resulting transcripts by identifying and grouping the issues and sub-issues that arose across all transcripts to construct issue maps. We explain how this was carried out with the example given in Figure 3. This shows that one of the RT problems that was repeatedly stressed was that of not being able to trace early project information. Through our analysis of the issue maps, we were able to identify what underlay this problem, examine what other issues were related to this problem, and further identify what was meant by the phrase "early project information". Although not shown in Figure 3 for clarity, we were able to decompose the component aspects of this problem in progressively more detail. In this way, we were able to uncover a broad picture of
the main problem areas underlying RT, and hence some high-level requirements for RT improvement.

**Figure 3:** Decomposition of one of the problems involved in RT. This was uncovered from the focus group sessions. This problem is the inability to trace early project information. The issues complicating this were found to be divided into those related to RE practice and those related to the characteristics of particular projects. Although this diagram only lists the component aspects of these issues, the focus group analyses enabled many of these to be understood further. For example, the reasons for a lack of ownership, the type of information that needs to be provided to delimit such ownership, and what further problems result from this lack of ownership.

### 2.4. Questionnaires

Following the focus groups, we made use of a two-stage questionnaire to target practitioners and to channel the subsequent data gathering more finely. These were piloted amongst members of the smaller company to help define their content and structure. The final versions were only distributed within the larger company. The questionnaires contained many types of question: (a)
factual questions to compartmentalise data; (b) opinion-seeking questions to obtain subjective data; (c) pre-coded questions to enable statistical, tabular and graphical analyses; and (d) open questions to encourage practitioners to volunteer data. Not all the questions were seeking information related to RT. Some were designed to categorise the respondents and some were designed to consolidate the answers given to previous questions.

The first stage of the questionnaire was short and contained ten general questions related to systems and software engineering practice. This was designed to mobilise response and to rapidly gather broad data from a wide population of practitioners involved in all aspects. It was also designed to target smaller populations from which more specific data could be gathered. Eighty of these questionnaires were distributed and sixty nine percent were returned. This questionnaire, the responses, and a synopsis of the analysed results, can be found in Appendix A.3.

The second stage of the questionnaire was longer and much more detailed. It came in many guises, as each was individually tailored to the primary working areas, job roles, and particular experiences of the practitioners responding to stage one. In this way, it was designed to elicit a deeper understanding of the actual problems and issues involved in RT and encountered in practice, and so to identify specific requirements for addressing them. The questions used in this questionnaire were taken from a reusable pool to enable some cross analysis of the responses. Thirty nine of these questionnaires were distributed and eighty five percent were returned. The questions, the responses, and a synopsis of the analysed results, can be found in Appendix A.4.

2.5. Interviews

We followed the questionnaires with two informal interview sessions. These were carried out with large groups of the questionnaire respondents and lasted one and a half hours each. They were used to corroborate the questionnaire answers, to probe beyond the answers to appraise their validity, to extract supplementary background information, to encourage spontaneous comments that are not possible in a questionnaire scenario, and to check the preliminary analysis of the replies. These interviews helped to substantiate the problem analysis and definition, as well as clarify the high-level requirements determined for RT support.

2.6. Observation and Participation

We also gathered data following the observation of, and participation in, a variety of RE exercises for different projects. This was carried out both within the large and small companies. The purpose of these were to analyse the dynamics of RE activities and so inform about requirements for supporting the RE process itself and for making this process traceable. For instance, we were able to observe groups of practitioners participating in Rapid Application Development (RAD) workshops. In such workshops, requirements are dynamically generated amongst a team of
stakeholders, but under the guidance of a facilitator. Both the process and its results are concurrently documented by a scribe. Whilst observing such workshops, we took comprehensive notes, and further collected any informal documents that were either individually or collectively produced. Our subsequent analysis was concerned with comparing these collected artifacts with the eventual end products of the workshop and investigating the numerous traceability issues involved.

### 2.7. Introspection

One of our activities involved iteratively constructing an RS for RT alongside the above investigations. This was carried out to assist with progressive problem description and so to drive the direction and focus of the data gathering activities. Our own RE experiences were continuously reflected upon, and informally documented, to help identify the problems that frequently arose and the requirements for supporting both this activity and its traceability. This was an invaluable personal aid for recognising and understanding the issues we began to uncover.

### 3. Problem Analysis and Definition

In this section, we present a synopsis of the main findings from our analyses of all the above data. It must be noted that we uncovered a wealth of interesting information, though we restrict our concern for the purpose of this thesis. In summary, we found the existence of three fundamental conflicts to underlie continued RT problems. These conflicts revolved around the lack of shared agreement concerning: (a) what RT actually is; (b) what causes RT problems in the first place; and (c) what RT should be able to do. We illustrate the reasons for each of these conflicts, discuss their implications for RT, and describe what needs to be done to address them.

#### 3.1. Lack of Common Definition

Since the introduction of the term "requirements traceability" by the U.S. Government's Department of Defense, each subsequent attempt at definition, either in the literature or by practitioners, has taken a slightly different form. We uncovered many definitions of RT which were either: (a) purpose-driven; (b) solution-driven; (c) information-driven; (d) direction-driven; else stated using a combination of the above.

##### 3.1.1. Purpose-Driven

Definitions falling in this category define RT in terms of what it should do. Examples include:

(1) Requirements traceability is "the means whereby software producers can 'prove' to their client that: the requirements have been understood; the product will fully comply with the
requirements; and the product does not exhibit any unnecessary feature or functionality" [Wright 1991].

(2) "Requirements traceability is the ability to adhere to the business position, project scope and key requirements that have been signed off" [Practitioner participating in a focus group].

3.1.2. Solution-Driven

Definitions falling in this category define RT in terms of how it should be implemented. Examples include:

(1) "Traceability refers to the ability of tracing from one entity to another based on given semantic relations" [Ramamoorthy et al. 1986].

(2) "Traceability refers to the ability to cross-reference items in the requirements specification with items in the design specification" [Roman 1985].

3.1.3. Information-Driven

Definitions falling in this category define RT in terms of the information that it should trace between. Examples include:

(1) "Requirements traceability is the ability to link between functions, data, requirements and any text in the statement of requirements that refers to them" [Practitioner participating in a focus group].

(2) The paragraph for requirements traceability must contain "a mapping of the engineering requirements in this (Software Requirements) Specification to the requirements applicable to this Computer Software Configuration Item in the System/Segment Specification, Prime Item Development Specification, or Configuration Item Development Specification" [DoD 1988b].

3.1.4. Direction-Driven

Definitions falling in this category define RT in terms of the direction in which it should be achieved. They tend to emphasise either forwards or backwards direction, or sometimes bidirectionality. Examples include:

(1) Traceability is "the ability to follow a specific item at input of a phase of the software lifecycle to a specific item at the output of that phase. The item may be software or documentation" [ESA 1987].

(2) Traceability enables "each requirement to be traced to its origin in other documents and to
3.1.5. Implications of Diverse Definitions of RT

RT problems can be seen to persist because no single definition accounts for all perspectives regarding what RT is. Each definition differs in emphasis and so delimits the scope of immediate concern. This situation has implications for the development and use of techniques, approaches, and tools for RT. This is because both the type and extent of RT that is provided will echo the definition of RT embedded in the technique, approach, or tool. The definition employed can thereby impose assumptions about aspects like the quantity and diversity of information that can be traced, the extent and longevity to which RT can be maintained, and even predefine specific techniques to use before having considered alternatives. The implications are especially pertinent in large projects. How can RT be consistently provided if each individual in a team, and each team in a project, have their own understanding as to what is meant by RT? Furthermore, how can techniques, approaches, or tools be coherently used together if they embed, and so support, incompatible notions of RT? In summary, we suggest that RT problems will continue to exist in practice so long as these dispersed viewpoints as to what RT is are not recognised and reconciled in some way.

3.2. Varied Sources of RT Problems

From our empirical investigations, it was evident that each practitioner had their own understanding as to the main cause(s) of the RT problem, along with what the corresponding effects are likely to be. These included:

(1) The existence of multiple incompatible and fragmented documents, from distributed sources, with no clear relationship to a unified RS.

(2) The inability to handle the increasing amounts of documentation.

(3) Change, and the slowness with which all its ramifications are taken into account, which leads to numerous versions of documents in various stages of evolution.

(4) The lack of an end-to-end RT process, plus the absence of a specified RT job description, thus leading to RT mismanagement.

(5) The involvement of too many, often uncooperative people, with inadequate expertise and individual agendas.

Our empirical finding with regard to the diverse causes of RT problems is also reflected in the literature. There, the fundamental problem underlying RT has been attributed to aspects such as:
Part II: Problem

3.2.1. Implications of Diverse Sources of the RT Problem

We found that many reasons were cited for the existence of RT problems. This is important to recognise because any particular technique, approach, or tool for RT will only ever focus on a subset of these more fundamental problems. This subset of problems will dictate the type of RT that each technique, approach, or tool provides, and will further influence what the perceived limitations with their RT are likely to be. In summary, we suggest that RT problems will continue to exist in practice so long as these multiple causes of the problem are not simultaneously recognised and addressed in some way, else where those factors most applicable to specific projects are not wholly accounted for.

3.3. Conflicting Needs for RT

From our empirical investigations, it was also evident that each practitioner had their own diverse expectations as to what RT should be able to achieve, and so as to the additional problems that improved RT should be able to relieve. These included:

1. The ability to establish control by managing multiple copies of documents and any iterations, changes, additions, deletions, and so forth.

2. The ability to enable the business case to drive the RE process so that each of its critical aspects is reflected in both the evolving RS and the end product.

3. The ability to provide access to original requirements and all their stakeholders, as well as to further highlight where intermediaries have been involved, and so improve both information sharing and the communication potential.

4. The ability to provide selective and filtered views of interconnected documentation to support different activities.

5. The ability to identify duplicated, conflicting, or non-verified requirements, then manage the transfer of these requirements between the different lifecycle phases.
Again, our empirical finding with regard to the diverse requirements for RT is also reflected in the literature. There, the expectations for RT have included aspects such as:

(1) The ability to promote a contractual approach to the development process [Edwards & Howell 1991, Stehle 1990].

(2) The ability to understand systems from multiple points of view and to assist in the pulling together of fragmented information [Easterbrook 1991].

(3) The ability to permit flexible process modelling and so support the evolvability of the RS [Fischer 1991, Johnson et al. 1991].

(4) The ability to offer some degree of assurance that specifications were written with user requirements in mind and so assist with user acceptance testing [Ramamoorthy et al. 1984].

(5) The ability to track requirements allocation, requirements flow-down between development phases, and the rationale and constraints used to develop product elements. This also includes support for the analysis of aspects such as consistency, completeness, test procedures, data integrity, safety, security, and change impact [Dorfman & Flynn 1984, Hamilton & Beeby 1991, Hoffnagle & Beregi 1985, Wilmot 1992].

3.3.1. Implications of Diverse Needs for RT

We found that the requirements for RT itself are diverse, contradictory, and often even changing. This is because RT is believed to be the answer to many additional problems encountered in systems and software practice, without recognition that it cannot singularly address all of the problems, and it is not necessarily the most appropriate solution for some. These findings imply that many practitioners claim to have experienced RT problems when they are unable to carry out those very activities that they expected RT could make possible. This is important to recognise because, any particular technique, approach, or tool for RT will be designed to support a subset of the above requirements and will therefore address some of these activities better than others. In summary, we suggest that RT problems will continue to exist in practice where these multiple expectations for RT are not simultaneously recognised and supported in some way, where these expectations are not managed and so remain inflated, else where the specific RT needs for a particular project or organisation are not well understood and so not taken into account by the support chosen.

3.4. Multifaceted Nature of the RT Problem

Techniques, approaches, and tools have been thrown at the RT problem without any thorough investigation of what the actual problem is that they should be dealing with. Through the
identification of the above conflicts, we can see that the RT problem is not perceived to be uniform. Those involved with requirements, design, implementation, testing, management, or the various other aspects in the systems and software engineering process each hold different opinions as to what RT is, why RT is needed, the problems experienced and attributed to RT, and the reasons for these problems. These findings imply that there is problem statement ambiguity as the phrase "RT problem" is being used to umbrella a collection of more fundamental problems. Complicating this is the observation that each practitioner's perspective with regard to these underlying problems is context-dependent and hence subject to change. Complicating this further is the observation that RT improvements are expected to yield the solution to changing, and often incompatible, additional problems.

Our high-level empirical findings therefore indicate that an all-encompassing solution to the RT problem is not likely. For this reason, it is essential to provide a framework through which we can identify the individual facets of the composite problem, and thus suggest ways to tackle these. With this objective we offer:

(1) A working definition of RT.
(2) A further analysis of the more fundamental problems underlying the RT problem.
(3) A specification of the basic requirements for improved RT.

4. Requirements Traceability Framework

In this section, we describe a conceptual framework through which we clarify the issues underlying the RT problem. We first provide a working definition of RT which also serves to partition the problem space. We then explain where the majority of the current support for RT lies and so suggest where the longer-term causes of RT problems are most likely to lie.

4.1. Defining RT

A working definition of RT needs to be general enough to encompass different views of RT, but specific enough to highlight its significant aspects. The definition we propose is derived from a definition used in a popular systems and software engineering standard and from a dictionary definition. Across all the definitions of RT that we identified, the definition which was most cited in the literature was that given in Box 5, taken from the ANSI/IEEE Standard 830-1984 [IEEE 1984].

"A software requirements specification is traceable if (i) the origin of each of its requirements is clear and if (ii) it facilitates the referencing of each requirement in future"
Part II: Problem

Box 5: A definition often used for the term "requirements traceability" (taken from [IEEE 1984]).

This standard further recommends:

(1) Backward traceability to previous development stages, which "depends upon each requirement explicitly referencing its source in previous documents" [op. cit.].

(2) Forward traceability to all documents spawned from the software requirements specification, which "depends upon each requirement in the software requirements specification having a unique name or reference number" [op. cit.].

The above definition states the conditions that must exist for an RS to be traceable, rather than providing a definition of RT itself. It emphasises backwards traceability from requirements in the RS to source documents and forwards traceability from requirements in the RS to later project documentation. The use of cross referencing techniques to achieve this RT amongst established project deliverables is also reflected.

To broaden the scope of this definition, to apply to RT itself and not suggest techniques to use and information types to make traceable, we make use of a dictionary definition of the word "trace", as shown in Box 6.

Box 6: What is meant by the word "trace" (excerpts from [Sykes 1978]).

The ability to "delineate" and "mark out" "perceptible signs of what has existed or happened" in the lifetime of a requirement to enable one to "pursue one's way along" this record.

Requirements traceability (RT) refers to the ability to describe and follow the life of a requirement in both a forwards and backwards direction (i.e., from its origins, through its development and specification, to its subsequent deployment and use, and through periods of on-going refinement and iteration in any of these phases).
4.2. Pre-RT and Post-RT

With reference to the working definition of RT in Box 7, we further suggest that RT itself can be divided into two basic types, which together provide a framework in which to locate and understand the nature of the underlying problems. These two types revolve around the written specification of requirements, the RS, or an equivalent document being used as a baseline. They are what we refer to as pre-requirements traceability (pre-RT) and post-requirements traceability (post-RT), defined in Boxes 8 and 9 respectively.

**Box 8: A working definition of the term "pre-requirements traceability".**

**Pre-requirements traceability (pre-RT)** refers to the ability to describe and follow those aspects of a requirement's life prior to its inclusion in the RS in both a forwards and backwards direction (i.e., requirements production and refinement).

**Box 9: A working definition of the term "post-requirements traceability".**

**Post-requirements traceability (post-RT)** refers to the ability to describe and follow those aspects of a requirement's life that result from its inclusion in the RS in both a forwards and backwards direction (i.e., requirements deployment and use).

Figure 4 shows the typical setting of RT to illustrate the distinction between pre-RT and post-RT. Although both types of RT are clearly essential, we emphasise this separation as our empirical investigations indicated that, in practice, RT problems are centred around the current lack of distinction between these two basic types. These two types deal with separate kinds of information, assist with different types of problem, and is only an implied distinction in some of the literature [Feather 1991, Mathews & Ryan 1989, Rzepka & Ohno 1985]. Therefore, the two main phases of a requirement's life, its on-going production and its deployment, were found to impose different requirements on potential support for its traceability. This means that comprehensive support for RT can only be provided through an explicit recognition of, and support for, their subtle differences and their seamless integration.
Figure 4: A simplified diagram to show the two basic types of RT, pre-RT and post-RT. For clarity, we only show vertical RT through a linear progression of subsequent documents. We ignore multiple subsequent documents, the many intermediate documents that are likely to be present, and horizontal RT between versions. Note the way in which requirements information is distributed and merged in successive representations. Note also the added complication of iteration and change propagation.

4.2.1. Support for Post-RT

Post-RT depends on the ability to trace requirements from and back to a relatively static baseline document, usually the RS, and through a succession of documents and products in which they are distributed. When changes are made to requirements in this baseline, they can be re-propagated through this chain of distribution. Post-RT therefore enables what has been referred to elsewhere as baseline management [Dorfman 1990]. Most of the existing support for RT, particularly from the commercial tools, is directed at providing post-RT.

The problems with post-RT mainly arise when the activities of those involved in RE deviate from the frameworks which have been set out for it to ensure RT. These generally presuppose top-down decompositional approaches to requirements extraction, expansion, and refinement. Accordingly, it has been suggested that the remaining problems experienced here are an artifact of informal development methods and could be eliminated by formal development settings [Finkelstein 1991b].

Even so, any further improvements with post-RT will only have a limited impact on reducing RT problems. This is because this type of RT does not reflect the fact that the baseline RS from which it operates is often but the end product of an on-going and exploratory process from which the
requirements placed into this baseline emerge. Post-RT deals with change to these requirements solely from the baseline. However, it is this exploratory process, plus the ultimate requirement sources, which need to be traceable to support such changes in the baseline and to assess their impact. This observation has recently been identified elsewhere as an important and a pressing challenge [Finkelstein 1991a, Harrington & Rondeau 1993].

4.2.2. Support for Pre-RT

In contrast, pre-RT is poorly understood and not comprehensively supported. Pre-RT depends on the ability to trace requirements from and back to their originating statement(s) and through their production and refinement process. In this process, statements from diverse sources, which are often conflicting and overlapping, are eventually integrated into a single requirement in the RS. Any changes proposed to requirements in this RS need to be both instigated and propagated from their source(s), and with reference to the production and refinement process, to indicate what in the RS and what elsewhere needs changing as a consequence. This is because seemingly unrelated requirements in the RS are often strongly interdependent, something which can only be observed where the process is rendered explicit. For example, if an organisational standard that was used to produce some of the requirements in an RS was subsequently changed, the identification of all the directly and indirectly affected requirements is problematic without pre-RT.

The existing commercial tool support, which is predominantly for post-RT, is not directly applicable for providing pre-RT. This is because most of these generally treat the RS or equivalent baseline document as a black-box and provide little to show that the requirements are only the end product of a complex process. As they rarely represent anything about this initial process, they provide no means to support the on-going and emergent nature of the work practices involved in producing and refining the RS. Instead, they tend to redefine rigid information categories for recording potentially traceable information and prematurely bind to requirements. Although we identified rudimentary ways in which pre-RT activities are currently attempted amongst the practitioners in our empirical studies, using informal notes and tagging schemes, we found the dominating technique to be a combination of head scratching accompanied by good memory.

4.3. A Need to Improve Pre-RT

We have stressed how the two types of RT, pre-RT and post-RT, are both crucial if we are to provide a comprehensive approach to RT. We have explained how post-RT is currently well supported and why the remaining problems here are not insurmountable. In contrast, we have mentioned that the issues that pre-RT are to deal with have not been well understood to date, so are not well supported. We therefore suggest that improvements in pre-RT are the most essential if we are to achieve any magnitude of improvement with RT. We frame our argument to support this in the context of Section 3.3 of Part I of this thesis where we argued that the prime importance of RT
lay in its ability to improve systems and software quality. To do this, we describe how advances in post-RT would only have a limited impact on quality, then describe why advances in pre-RT would provide more far-reaching improvements.

### 4.3.1. Limited Impact of Improvements in Post-RT on Quality

Although there is much support for post-RT, systems and software still frequently fail to attain the levels of quality anticipated for them. This is because post-RT support only deals with those phases of a requirement’s life which result from its specification in the RS and so embeds the assumption that the requirements in the RS are relatively easy to obtain, accurate, and stable. This is an assumption that is echoed where "quality" is defined in terms of meeting the specified user and quality requirements.

These assumptions concerning the nature of requirements are often misguided. For a start, user requirements are notoriously difficult to obtain with any accuracy, are frequently unstable, and become redefined over time. Furthermore, user satisfaction tends to be a collective and subjective matter for which reliable measures cannot be easily articulated. Secondly, quality requirements are commonly imported from external standards or policy documents, such as that described by [ISO 1991], where they are generally pre-specified along with metrics for measuring compliance [Keller et al. 1990]. Due to the sheer number of potential quality attributes in such documents, [Boehm et al. 1978] provides a representative list, these need to be individually adopted and tailored on a project-specific basis [Buckley & Poston 1984]. Furthermore, definitions of quality change, the relevance of metrics to quality change, and quality requirements and metrics are often actively constructed as the phases of a project progress. The very nature of these user and quality requirements indicates that, although post-RT can promote concern for quality during development and help to assess subsequent conformance to these requirements in the RS, it cannot guarantee quality. It lacks the means to probe beyond what is specified as a requirement, re-work these requirements if needed, and so does not account for all phases of a requirement’s life.

### 4.3.2. Potential Impact of Improvements in Pre-RT on Quality

We argue that additional improvements in quality could be obtained if the quality-oriented approaches to systems and software development included comprehensive support for the production and refinement of requirements in conjunction with that offered for their deployment by post-RT support. This is because pre-RT enables previously closed issues, even decisions concerning how to conduct RE itself, to be made explicit, possible to re-open, and possible to re-work. Quality improvements could therefore be attained through pre-RT’s potential to assist issues like:

1. The handling of changing requirements [Bersoff & Davis 1991].
Part II: Problem

(2) The ability to make logical sense of the requirements and prioritise them [Short 1988].

(3) The auditing of requirements to learn about best practice [Chikofsky & Rubenstein 1988].

(4) The repeatability of this best practice [Jarke & Pohl 1992].

In a critique of quality assurance activities in software engineering, Loka reports how most of these activities tend to be carried out towards the end of a project, after the original development team has moved on [Loka 1992]. Accordingly, any defects that are identified are inadequately addressed out of context, by other parties, and by directly weeding them out of the end product itself [Flatten 1992]. This situation leads to ever-deteriorating quality. The benefit of pre-RT is that it would enable a quality culture to be built in from project conception and for quality to be an on-going concern. This is essential because a positive correlation has repeatedly been stressed in the literature between the earlier that support is injected into the development process and the quality of the software that can be delivered [Palmer & Fields 1992].

It is also important to mention here that improvements in pre-RT would also offer the potential for greater economic leverage than further improvements in post-RT. This is because a significant proportion of development and maintenance cost, time, and effort is presently spent in compensating for requirements invisibility [Devanbu et al. 1991]. To make use of an RS, as well as to maintain it, it is often necessary to reconstruct and rediscover an understanding of how it was produced. This is regarded as a complex and error-prone endeavour in practice.

4.4. Pre-RT will be Responsible for Long-Term RT Problems

Unlike post-RT, it has been argued that the problems of pre-RT will remain, irrespective of formal treatment [Finkelstein 1991b]. As the production phase of a requirement's life is inherently paradigm-independent, pre-RT is likely to be the only contributor to RT problems in formal transformational settings. Our empirical findings intensify this concern. They strongly indicate that the majority of the underlying problems currently attributed to poor RT are due to the lack of pre-RT or to inadequate pre-RT. Therefore, we argue that techniques to record and trace the information related to the production and refinement of requirements in an RS will be the most instrumental in reducing RT problems in the longer-term.

5. Dominating Issues in Pre-Requirements Traceability

Having explained how insufficient pre-RT is likely to be the main contributor to long-term RT problems, especially in formal development settings, our problem definition and requirements gathering exercises were re-focused to determine what improvements in pre-RT would involve and how these could be realised. In this section, we first outline why the main problem compounding pre-RT improvements is informational in character, since pre-RT problems will continue to exist
Part II: Problem

where practitioners are unable to trace the information about requirements production and refinement that they require. We then describe how improvements basically rely on appeasing the main parties involved. In the light of these issues, we outline the key requirements for providing and using pre-RT, and then go on to suggest some possible solution options.

5.1. Missing Information

Traceability is really only a technique for filtering and accessing information. In the case of RT, this is information about requirements. In the case of pre-RT, this is more specific information about requirements production and refinement. The empirical data we collected indicated that the majority of the RT problems attributed to pre-RT are informational in character as problems mainly arise when practitioners are unable to trace the particular information they require about the production and refinement of requirements in the RS. With the present emphasis on developing formalisms in which to describe elicited requirements, to promote transformational approaches, there is a natural tendency to black-box what is elicited and thus compound the problem. An obvious first step towards addressing these informational problems involves obtaining and recording comprehensive details about the RE process and organising these details so that they are traceable in multiple ways.

5.2. Barrier Imposed by a Provider/End-User Conflict

Our empirical investigations clearly indicated that the main barrier confronting this obvious first step is what we refer to as the provider and end-user conflict. By this, we mean that the two main parties involved, those who would be in a position to make pre-RT possible and those who would subsequently require pre-RT to assist their work, have conflicting problems and needs. One party's benefits are often obtained at the other party's expense. Addressing one party's concerns often makes it problematic to address the other's.

The questionnaires pointed out many such conflicts. For instance, those involved with the design, implementation, maintenance, and managerial aspects of a project, were found to attach a high priority to the ability to trace back to why requirements in the RS are requirements and to process information about RE. In contrast, those involved in writing the RS and those involved in the work conducted prior to this, attached no such importance to this same ability. Amongst their reasons for this lack of concern included: (a) being fully aware of such information and not believing it to be of relevance to others; (b) having too little time or support for providing yet more documentation, which would anyway distract from their main project tasks; and (c) the belief that it is unlikely that all those involved in a project would be equally committed to providing for this facility, so feeling there to be little point in their own individual efforts. This finding alone indicates that those in a position to provide for pre-RT have a low motivation to do so, even though other parties involved in the development process demonstrably required it. This is often
even the case when the same individual assumes both positions. Not surprisingly, further questionnaire responses from the same individuals revealed that much of their time is actually spent in explaining why things are requirements to others involved in later phases. They also revealed that many of those involved in these later project phases often spent much of their time actively locating, usually unsuccessfully, those individuals involved in the RE process who could provide such information.

5.2.1. Problems Faced by Providers

Drawing from the empirical data we collected, the main problems faced by the potential providers of pre-RT are listed below. These are then summarised in Figure 5.

1. Pre-RT is perceived as an optional extra by those in a position to resource it, with post-RT being given higher priority. Therefore, insufficient time, personnel, and resources are allocated to its provision. It is often only when a requirement is rendered problematic, which appears to occur with increasing frequency as a project progresses, that the need is felt for making accountable the requirement-producing activities.

2. Pre-RT can rarely be achieved by uncooperative individuals, as individual efforts are typically ad hoc, localised, and not coordinated, especially where there is an imbalance between the extra work involved and the personal benefits gained. Pre-RT needs to be a combined and full-time responsibility by all involved in the RE activities to fully succeed. An explicit allocation, awareness, and management of the different roles that practitioners need to assume to achieve four interdependent tasks of obtaining, documenting, organising, and maintaining the required information, is typically absent.

3. A shared understanding of the diverse requirements for pre-RT, those imposed by different stakeholders throughout a project's life, is generally lacking. Also, accounting for the unique and unpredictable nature of these end-users' requirements is problematic. Therefore, there is an obvious tendency to focus only on the visible needs of immediate end-users.

4. Concern for pre-RT diminishes, and concern for post-RT increases, after the first snapshot at an RS has been formally signed off. RS production and refinement is a social and on-going activity for which concern must continue throughout a project's life. Maintenance of this concern is problematic as the exact nature of this activity cannot be fully predefined and there tends to be poor feedback of later work. Requirements changes are generally not reflected here.

5. The requirements production and refinement information required to be made traceable cannot always be readily obtained and documented, as is the case with tacit knowledge or confidential rationale. That information which is documented, particularly rationale, varies
widely in quality and is dependent on factors like time constraints. Furthermore, the predominance of a deliverable-driven culture in industrial settings can actively discourage the gathering of certain information which does not fit into predefined deliverables.

(6) Documenting information about requirements production and refinement does not mean that it will be subsequently traceable, unless it has been explicitly organised for that purpose.

(7) Requirements production and refinement information that is structured so that it can be traced in many ways itself offers no guarantee that it will be up-to-date. There are often problems in accounting for all the possible changes affecting this information. This is mainly due to an immature change culture and the cyclic dependency upon RT itself.

(8) Poor feedback regarding best practice and little dedicated support, be this clerical, procedural, or computer support, simply perpetuates the problems that providers face.

![Figure 5: The main problems underlying the provision of pre-RT. Note that these could also be said to apply to the provision of RT in general.](image)

5.2.2. Problems Imposed by End-Users

Again, drawing from the empirical data we collected, the main problems imposed by the potential
end-users of pre-RT are listed below. These are then summarised in Figure 6.

(1) A stereotypical end-user cannot be predefined, so requirements for end-use of pre-RT will differ, and even be inconsistent for a specific individual.

(2) The potential quantity and heterogeneity of requirements production and refinement information required by all possible end-users precludes total predefinition. Personal contact is always heavily relied upon because much of what is later required is often undocumented, inaccessible, outdated, or documented in a form not suited to the use for which it is required.

(3) The various ways in which end-users require access to possibly combinations of requirements production and refinement information, along with how this information is most suitably presented for their purposes, also cannot be predefined.

(4) Each end-use situation exhibits unique pre-RT requirements. Problems will always exist if end-users do not have the ability to filter and access the different types of information about the production and refinement of requirements that they want to trace under different circumstances.
Figure 6: The main problems imposed by the end-users of pre-RT. Note that these could also be said to apply to the end-use of RT in general.

5.3. Key Requirements for Pre-RT

To provide an exhaustive account of each requirement in an RS, technological solutions can readily assist with aspects such as the documentation, organisation, and retrieval of huge amounts of information. However, technological solutions do not offer the complete answer, as fundamental problems will still reside with the people involved. The real challenge for improving pre-RT therefore lies in the ability to address the problems confronted by the two main parties. The end-users need to be able to predefine their anticipated requirements for pre-RT and make these clear to the providers. The providers need to be able to identify relevant information and document it in a (re)usable form to suit the anticipated needs of all end-users. Furthermore, this must be achieved almost as a by-product of what is considered mainstream work, entail some immediate benefits, else be given a much higher priority and explicit support.

Our investigations therefore led to the production of a comprehensive RS for pre-RT, given in Appendix C.1. This was developed in conjunction with practitioners and subsequently distributed.
to practitioners in [Gotel 1992]. Here, we only focus on clarifying what we consider to be the key requirements to provide and make use of pre-RT in Tables 3 through to 7.

### 5.3.1. Fundamental Requirements

In Table 3, we highlight the high-level requirements to provide and make use of pre-RT. We list the four most fundamental problems complicating improvements and suggest what is needed to alleviate them. The first three mainly relate to provision and the last to end-use. We do not explore exactly how these requirements could best be met here.

<table>
<thead>
<tr>
<th>Problems</th>
<th>Requirements</th>
</tr>
</thead>
</table>
| 1. Lack of "buy-in". | 1.1. Education in the need for and importance of pre-RT.  
1.2. Project time dedicated to providing pre-RT.  
1.3. Assistance for the routine and mundane aspects involved, perhaps using clerical or tool support, to reduce the manual effort.  
1.4. Up-front commitment from all involved.  
1.5. Benefits and incentives for all involved. |
| 2. Not a dedicated job. | 2.1. How pre-RT is to be achieved on a project needs to be established early on.  
2.2. Provide visibility of organisational structures, work roles, and responsibilities.  
2.3. Establish continuity, through common threads of involvement, perhaps using facilitators or documentation control managers. |
| 3. Uniqueness of each situation in which pre-RT is to be provided. | 3.1. Support for flexible working practices of practitioners, like method independence, so the ability to provide pre-RT is not dependent on having followed predefined methods.  
3.2. Support for the coordination of collaborative and cooperative aspects of working.  
3.3. Ability to capture details about the process of requirements production and refinement in documentation.  
3.4. Ability to re-organise and re-define information.  
3.5. Support for information sharing and distribution, preferably through documentation, with visibility of document ownership. |
| 4. Uniqueness of each situation in which pre-RT is to be used. | 4.1. Awareness by all involved in a project of some of the typical and exceptional informational requirements for end-use.  
4.2. Ability to identify and account for the required manner of use regarding end-user access to, and presentation of, trace information.  
4.3. Awareness of the implications imposed by different user, project, and task characteristics in situations of end-use, plus the ability to |
5.3.2. Requirements for End-Use

In Table 4, we highlight what is basically needed to make use of pre-RT, so focus on problems in the ability to access and trace required information about the RE process. We list four basic problems faced by those who would make use of pre-RT and make suggestions as to what is needed to address each of these issues.

<table>
<thead>
<tr>
<th>Problems</th>
<th>Requirements</th>
</tr>
</thead>
</table>
| 5. Restricted end-use, as types of possible trace are generally predefined. | 5.1. Awareness of potential types of pre-RT required.  
5.2. Flexibility of access to documented information.  
5.3. Ability to filter documented information in different ways.  
5.4. Ability to dynamically create traces on-the-fly. |
| 6. End-user is unaware of all the possibilities open to them. | 6.1. Visibility of documented information available to be traced.  
6.2. Visibility of predefined or example traces.  
6.3. Visibility of ways in which both canned and user-defined traces can be requested and invoked. |
| 7. Problematic for end-user to request types of pre-RT required. | 7.1. Flexibility in the manner in which pre-RT can be requested and invoked.  
7.2. Assistance in setting up views, filters, and access mechanisms to documented information. |
| 8. Documented information that is traced is unusable or unsuited to end-user needs. | 8.1. Rapid retrieval and presentation of information that is traced, according to user-definable criteria.  
8.2. Ability to represent the same information in multiple ways simultaneously. |

Table 4: Basic requirements for end-use of pre-RT.

5.3.3. Requirements for Provision

In Tables 5 to 7, we separate and describe the basic requirements needed to address the main problems facing pre-RT provision. These are the problems of obtaining, recording, and then organising and maintaining information for pre-RT purposes.

5.3.3.1. Obtaining Information

In Table 5, we highlight what is basically needed to obtain the RE information that end-users want
to be able to trace using pre-RT. We list three fundamental problems which complicate the issue of obtaining required information and make suggestions as to what is needed to address each of these issues.

<table>
<thead>
<tr>
<th>Problems</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>9. Large amounts of information produced.</td>
<td>9.1. Awareness of information relevance and priorities, like what information is crucial, desirable, or superfluous for a project.</td>
</tr>
<tr>
<td>10. Not all information that is required is made explicit.</td>
<td>10.1. Support to make more information about RE explicit. 10.2. Emphasis on open communication, improved visibility, clarity of detail, and ownership. 10.3. Production of RE process information as a by-product of working practices, through automated support, clerical assistance, or the like.</td>
</tr>
<tr>
<td>11. Representative information often not available.</td>
<td>11.1. Ability to obtain more objective and detailed information. 11.2. Assistance for the immediate production of information whilst “ready-at-hand”, so whilst experiences are still recountable.</td>
</tr>
</tbody>
</table>

Table 5: Basic requirements for obtaining information to provide pre-RT.

5.3.3.2. Recording Information

In Table 6, we highlight what is basically needed to record and document the RE information that is obtained for pre-RT purposes. We list four fundamental problems which hinder the accurate and useful recording of this information and make suggestions as to what is needed to address each of these issues.

<table>
<thead>
<tr>
<th>Problems</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>12. Loss of information or change in emphasis.</td>
<td>12.1. Ability to directly document information obtained at the time it is produced. 12.2. Ability to directly document information obtained in its original form, be this informal, unstructured, incomplete, in different languages or media, and so forth.</td>
</tr>
<tr>
<td>13. Predefined documentation formats designed for specific types of information.</td>
<td>13.1. Flexibility of documentation format, so the ability to provide pre-RT is not dependent on having followed specified methods and having produced information to fit predefined document templates.</td>
</tr>
<tr>
<td>15. Documented</td>
<td>15.1. Ability to define linkages between documents, for traceability</td>
</tr>
</tbody>
</table>

90
Table 6: Basic requirements for recording information to provide pre-RT.

5.3.3.3. Organising and Maintaining Information

In Table 7, we highlight what is basically needed to collectively structure and organise the individual pieces of documented RE information for pre-RT purposes. We list six fundamental problems confronted when attempting to organise and maintain this information and make suggestions as to what is needed to address each of these issues.

<table>
<thead>
<tr>
<th>Problems</th>
<th>Requirements</th>
</tr>
</thead>
</table>
| 16. Individual documents are often incompatible, so hard to structure consistently. They are frequently of different types, at different levels of granularity, and so forth. | 16.1. Support to get documents into a usable and reusable form, so they are generic, abstract, and traceable.  
16.2. Support to compartmentalise documents as modular viable systems, so they are self-contained.  
16.3. Assistance with integration, alignment, levelling, composition, decomposition, partitioning, merging, and so forth, of documents.  
16.4. Support for the definition and management of document details on a project-specific basis, like whether duplication is allowed, the levels of granularity to use, and so on. |
| 17. Uncoordinated structuring and organisation. | 17.1. Central availability, as far as possible, of all documents produced.  
17.2. Global visibility of all fragmented documents, document interfaces, and document structures being defined.  
17.3. Early and on-going visibility of all documents produced and their document interfaces.  
17.4. Assistance for handling the organisation and re-organisation of documents, either through dedicated roles, principles, or automation. |
| 18. Distributed nature of information across documents, so hard to change information, and hard to manage the effects of change. | 18.1. Support for the formulation of tightly cohesive individual documents, to control duplication, localise detail, and so forth.  
18.2. Establish well-defined document interfaces which define attributes such as the changeable and non-changeable parts of a document.  
18.3. Ability to change information and control these changes through document interfaces. |
| 19. Rigidity of structure, so information is accessible in set ways, thus providing | 19.1. Flexibility to enable information to be rapidly structured and re-structured in multiple ways, to enable different types of traceability, views, and filters.  
19.2. Support for multiple and flexible link types between documents, |
single views and making them unwieldy to change. to enable new and alternative structures.

19.3. Ability to define set paths through the information, with the added ability to dynamically define new paths.

Problems (cont...) Requirements (cont...)  

20. Changes made to documents are often too slow, not visible, so out of date information is often acted upon.  

20.1. Assistance for rapid change, like the automatic handling of change details, explicit monitoring for change, and so on.  

20.2. Ability to generate change documentation.  

20.3. Immediate distribution of changes, their repercussions, and change documentation to all involved or affected.  

20.4. Regular conformance checks and quality assurance reviews to check that information is up-to-date, that required information is present, that pre-RT is possible, and so forth.  

21. Undesirable and unidentifiable repercussions of change.  

21.1. Ability to trial and explore change, prior to making the actual change, through the provision of localised and global views, visual demonstrations, or the like.  

21.2. Ability to identify forward and backward impact of change.  

21.3. Ability to undo.  

21.4. Information persistence.  

| Table 7: Basic requirements for organising and maintaining information to provide pre-RT. |

5.4. Addressing Informational Issues

Having outlined the key requirements for improving pre-RT, based on the problems we identified through empirical studies, we now show how many of these can be tackled by much of what already exists in commercial and academic arenas. In particular, we describe how the basic requirements for pre-RT provision can be met by strategies which increase the awareness of information to obtain, techniques which obtain and record such information, and mechanisms which keep such information up-to-date and accessible. We further describe how the basic requirements for pre-RT use can be met by exploiting some of the developments in information retrieval, artificial intelligence, and human-computer interaction.

5.4.1. Increased Awareness of Information to Collect for Pre-RT

Although studies have been carried out to reveal what project information is required by those involved in the different phases of development, to inform what information needs to be collected [Kuwana & Herbsleb 1993], our investigations strongly indicate that it is not possible to generalise such findings. The appropriate amount and type of information required about RE will remain subject to controversy. This can range from deliverables only, through all explicitly generated information, to an unbounded quantity of implicit information deemed essential for defining the
context. Perhaps simple dramaturgic models would be the most useful to promote awareness of the types of information that should be collected about RE, capturing: (a) the act, or what was done; (b) the scene, or when and where it was done; (c) the agent, or who did it; (d) the agency, or how they did it; and (e) the purpose, or why they did it [Burke 1969].

It is common to tackle the issue of collecting information to assist focused activities by categorising the type of information required. For example, the Decision Representation Language (DRL) of the SIBYL system [Lee 1990] and the argumentation scheme of the gIBIS system [Conklin & Begeman 1988], both provide explicit frameworks for representing the structure of decision making and design deliberation. They increase the awareness of the information that is needed to support particular activities and so, in turn, inform what needs to be captured. However, such schemes have been found to artificially restrict user input, emphasise the structure as opposed to the content of the information, and each be best suited to particular domains, systems, and tasks. Nevertheless, the current research into the development of RT models, as described in Part I of this thesis, is probably of most interest here. The intention behind such modelling is to increase awareness of the needs of various stakeholders in the development process by guiding the different types of information and relations that should be maintained for RT purposes. However, as the recommended categories will not always be shared by all those involved in a project, or by different projects, their use will likely be prone to individual subjectivity. Some of these problems could be assisted by establishing dedicated job roles to provide for RT according to these models.

5.4.2. Obtaining and Recording Information for Pre-RT

To tackle the problems involved in obtaining and recording information for pre-RT purposes, we could take advantage of the progress which has been made in the ability to obtain and record diverse sorts of project-related information. For instance:

(1) The history of the requirements evolution process, as in the REMAP model for the representation and maintenance of process knowledge [Ramesh & Dhar 1992].

(2) The history of RE artifacts, the decisions made in the RE process, and the "ways-of-working" through this process [Moreno et al. 1994, Rolland 1994a, Rolland 1994b].

(3) Requirements trade-offs, as in the KAPTUR system [Bailin et al. 1990].

(4) Explanations and justifications, as in the XPLAIN system [Neches et al. 1985, Swartout 1983].

(5) The design rationale of teams, as in the Questions Options and Criteria framework [MacLean et al. 1991].
Part II: Problem

(6) A record of collaborative activities, as in the ConversationBuilder tool and the OMNI model [Kaplan 1990, Lange et al. 1994] and, more specifically, of discussions about requirements, as in the Inquiry Cycle [Potts et al. 1994].

(7) The tangible products produced and used, the settings in which they have been developed and maintained, and the processes carried out upon them, as in the Intelligent Software Hypertext System (ISHYS) [Garg & Scacchi 1989].

(8) The enterprise or business environment for which systems are being built [Yu & Mylopoulos 1994].

The requirement for gathering more information about RE for pre-RT purposes could be met by amalgamating support for the above into an exploratory workbench or requirements pre-processor. Additional advances in the ability to obtain and represent information about the tacit aspects of the RE process could be gained from work examining the problems of identifying and representing information about organisational issues in requirements analysis [Jones & Brooks 1994] and ongoing work directed towards the use of ethnography and ethnomethodology to inform requirements gathering [Hughes et al. 1995, Jirotka 1991, Jirotka et al. 1995, Luff et al. 1993, Sommerville et al. 1993]. Such work would be equally useful for studying and describing information related to the working practices and organisation of those involved in the RE process itself.

5.4.3. Organising and Maintaining Information for Pre-RT

Support for the progressive development of project information, and any subsequent changes to it, requires flexibility of information content and structure, plus sophisticated change mechanisms. There is relevant work in many areas for addressing these kinds of issues, like through the use of:

(1) Modularisation techniques for organising RE documents [Parnas et al. 1985].

(2) Viewpoints as a principle through which to organise RE information [Finkelstein et al. 1992, Lowe 1985].

(3) Logical frameworks to model an RS in such a way as to support its gradual elaboration [Dubois 1990].


(5) Abstractions which support rigorous reasoning about change [Ward 1992].

(6) Version control and configuration management techniques to assist change control [Bersoff
Part II: Problem

(7) Hypertext or hypermedia to provide explicit visibility of structure and to maintain relations [Garg & Scacchi 1989, Wood et al. 1994].

(8) Boundary objects to indicate how RE information could be structured so that it could be shared between dispersed individuals, for different purposes, and so be amenable to multiple forms of trace [Star 1989].

Any necessary extensions in all these relevant areas would primarily be to deal with the handling of more informal and unstable information. Other improvements could result from the introduction of explicit job roles, supported by suitable tools and techniques, to cover the responsibilities of: (a) project librarian, to collect, clean-up, and distribute information; (b) information base manager, to coordinate, control, and ensure RE information is of quality and up-to-date; and (c) traceability facilitator, to establish and ensure the traceability of all RE information.

5.4.4. Flexible Access to and Presentation of Information for Pre-RT

The current potential for RT remains predominantly hardwired [Flynn & Dorfman 1990]. This is because providing the right amount of desired information, at the level of detail necessary for the problem at hand, is a problem that is not unique to RT [Bocker & Herczeg 1990]. For this reason, many developments in information filtering and information retrieval, like adaptive systems based on fuzzy logic or user modelling, are directly applicable and can address many requirements here. Much work in areas such as artificial intelligence, human-computer interaction, and knowledge-based systems, is also highly applicable. [CACM 1992] provides only a taster of such potential. Amongst the RE literature, recent work separating the internal representation of requirements information from its flexible presentation is also worth mentioning [Johnson et al. 1992].

Such advances mean that programmable multimedia workstations for end-users of pre-RT are recommended. Amongst many benefits, these could enable: (a) the retrieval of multimedia information through graphical and textual traces; (b) diverse ways to visualise traces, which could assist activities like impact analysis, say by presenting requirements dynamically using animation techniques or with links that light up when traversed; (c) concurrent global and local traces; and (d) alternative engaging methods of interrogation to define pre-RT information requirements in the first place. Together, the aforementioned areas could be exploited to provide flexible and user-definable pre-RT on-the-fly, enabling traces which dynamically mature to queries and end-user situations.
6. Crux of the Requirements Traceability Problem

In this section, we argue why long-term RT problems will not be eliminated, despite improvements in the above areas which could help address those outstanding informational issues that are crucial for attaining pre-RT improvements. This is because there will always be situations where practitioners will be unable to trace the information they require. We provide many reasons for this state of affairs and explain how this is currently dealt with when encountered in practice. As a consequence, we then argue that the crux of the pre-RT problem, and in turn of the wider RT problem, is the inability to locate and access the human source(s) of requirements, requirements-related information, and requirements-related work. We strengthen this argument with data from our empirical studies, which highlights both the causes and consequences of this inability, and with a pointer to related findings by others. Also drawing from our empirical data, we summarise some lessons from best RT practice which could be useful in tackling this more specific problem, and then suggest a possible way forward.

6.1. Information is Still Missing

As seen in the previous section, most of the support that can currently be provided for improving pre-RT mainly focuses on developing comprehensive repositories of project information related to RE, and then providing elaborate mechanisms through which to selectively access and present this information. However extensive or sophisticated the advances here become, we argue that long-term RT problems will not be completely eliminated. This is because our empirical studies strongly indicated that, even with the use of detailed pre-RT models, there will always be situations in practice where the particular information a practitioner wants to trace about requirements will either: (a) not be there; (b) be tailored to a different audience; or (c) not be entirely suited to the purposes at hand. Therefore, practitioners will still claim to have RT problems.

Many reasons can be provided as to why this will continue to be the case, including:

(1) The requirements that any individual or group has for pre-RT are situation-specific and not amenable to complete predefinition. That information which is potentially relevant to assist later project phases can rarely be accurately or exhaustively predefined early on in a project. Moreover, it is impossible to know a priori all the possible uses to which this information will be put, so there will always be problems when information is encountered for purposes for which it was not intended.

(2) Information generated as a by-product of enforced adherence to methods, or process models, will also vary in reliability. This is because it is unrealistic to assume that these will always be used as intended [Parnas & Clements 1986]. Manually provided information will suffer
from subjectivity and incompleteness, not only because it is difficult to be reflexive, but because notions of relevance differ, classification schemes are rarely shared, and equal commitment to detail is unlikely [Ehn 1988]. This is particularly the case when the purpose of the documented information is to rationalise action [Giddens 1976]. Furthermore, just as it has been found that the reporting on any decision making carried out in organisational settings almost invariably deviates from any guidelines set out for it [March 1991], information about RE work and RE experiences is similarly unlikely to fit into the neat forms or templates set out for it. Such information is generally incomplete, fragmented, incrementally captured, and often tacit.

There are various contextual factors that can be drawn upon and made relevant in documentary practices. This means that accounts are highly selective and that different accounts can be provided of the same event [Buttny 1993, Goffman 1971]. More specifically: "... each utterance or physical doing that the individual contributes to a current situation will be rooted in his biographical, personal identity" [Goffman 1974]. For such reasons, accounts need to be open to further reinterpretation, and even to possible repair. Buttny suggests that, rather than having a single documented account, there is a need to provide a link to all the people involved to enable multiple accounts: "The labeling of an incident and the ascription of responsibility for it is not enacted by a distant, neutral observer or judge, but by interactants variously positioned and aligned in social contexts. We have no recourse but to consider actors' accounts because the meaning and significance of events is a consequence of such discursive, communication practices" [op. cit.]. Furthermore, as artifacts are inherently incomplete when isolated from the motivations and strategies for their creation and use, it has been suggested that the best possible record has to be a complete photocopy of an event [Raffel 1979].

There are also many contextual factors that can be drawn upon when interpreting a piece of documentation: "... what is in the record in the first place and what is drawn from it in decision making are determined by the values, tastes, ideologies and biases of those who contribute to and draw from the record system" [Wheeler 1969]. Absence of this information is a barrier to understanding, as: "The understanding of a text consists first of all placing it in its proper socio-political configuration" [Said 1979].

In many projects, information which is classified as confidential, often provides essential rationale for requirements. However, there are often elaborate access control schemes in place to either prevent or enable degrees of access to such information, and disclosure of such information is generally non-negotiable.

As an aside, studies of medical consultations have reported that problems in understanding patient case histories typically arise when doctors have written too much information [Heath 1982].
Within this domain, it has been found that economic descriptions are crucial, and further need to be based on shared codes and established practices. Drawing from these observations, will the strive for gathering excessive information about the RE process similarly inhibit understanding? If so, what descriptions, codes, and practices would be most appropriate to avoid this in the RE domain?

6.2. Dealing with Missing Information

Considering the above, it is hardly surprising that the most significant findings from our empirical analyses were:

(1) The lack of agreement regarding the quantity and type of information that practitioners wanted to trace about requirements.

(2) The extreme importance that practitioners attached to personal contact and informal, often face-to-face, communication with appropriate participants in the RE process.

It was evident that, due to (1) above, practitioners regularly encountered circumstances in which required information was absent. When faced with this situation in practice, we found that practitioners all resorted to the same fall-back strategy. This fall-back strategy involved identifying and talking to those individuals or groups who could provide the required information or fill in any of the missing details. These tended to be those who were generally responsible, in one way or another, for the information or work in question. However, the second finding was not just found to be a consequence of the first, to cope with those inevitable situations in which required information was absent, but even when required information was present this fall-back strategy was found essential to help account for the situated character of information needs. This was because it enabled any information that was available to be consolidated, supplemented, questioned, validated with the appropriate stakeholders, and for any remedial work to occur.

So, although we obtained no consensus amongst practitioners regarding the information they wanted to trace about requirements, the most significant finding was agreement that the most useful pieces of information were:

(1) The ultimate source of a requirement, meaning the individual(s) whose requirement it is.

(2) Who was involved in the various activities that led to a requirement's inclusion in an RS, meaning the source(s) of any additional information generated as a consequence of the RE process.

What these findings indicate is that both eager and lazy generation of information is required for pre-RT purposes. By eager, we mean the documentation of requirements-related information
whilst actively engaged in RE. Such information is often well suited to the immediate and short-term needs of those involved and useful as a later reference point. With time, such a static snapshot may be less suited to additional needs and is increasingly difficult to interrogate if it is the only information recorded. In contrast, information generated lazily and on need by those originally responsible, can be provided with the benefit of hindsight and further targeted to suit specific needs. However, without reference to information recorded at the time, to regain some context, such information would become increasingly difficult to reproduce over time. As pointed out by [Bolter 1991], this need to combine the permanence of writing with the flexibility of conversation can be seen as far back as in the works of Plato.

Regardless of major technological advances, this fall-back strategy for the lazy generation of RE information will always be desirable, and in many cases it may well be essential. Practitioners therefore require access to accurate information which informs them about the human source(s) of any requirements information that is recorded. This reflects the fact that people are often the final authority about requirements and, as such, are frequently able to prevent potential RT problems. This also corroborates a growing awareness that some of the most vexing problems confronting RE are in fact social, political, and/or cultural in nature [Goguen 1993].

6.3. Problems in Practice

The most outstanding information that needs to be made traceable is therefore that which serves to identify those who have been involved in the various aspects of RE. However, the ability to locate such individuals or groups, or suitable alternatives, was reported to be problematic in practice. Drawing from our empirical data, we describe why this is currently problematic to do, and go on to list some of the knock-on problems that this causes. We then mention related work which has indirectly pointed to this issue.

6.3.1. Causes

To date, there has been little focus on strategies for recording and maintaining details about RE participants. This situation reflects an absence of concern in typical RE standards, guidelines, and methods. It is therefore unsurprising that, in practice, the end products of RE are generally divorced from details of those who originally produced the requirements and from those who were involved in all phases of their refinement. Information about RE participants is therefore either not available, outdated, or refers to those who wrote the documentation as opposed to those who formulated the content. This situation arises because the predominant ways to attach details of participation in the process are either through a document field labelled "author" or through the use of annotation mechanisms. Such records are generally inadequate for identifying participants because:
Part II: Problem

(1) They are often based on coarse and static notions ownership. These have often been derived from predefined models of responsibility for a project, whereas such notions are subject to continuous change and are often only transient.

(2) They can encourage singular notions of ownership, whereas requirements production and refinement is an inherently social accomplishment, so requirements tend to have a variety of simultaneous sources and people involved in their emergence in different ways.

(3) They generally remain obscure, hidden in documents fields, so are rarely amenable to any form of analysis.

(4) They do not tend to provide a suitable basis through which to represent changing patterns of participation as either the information content or the organisational constituents evolve.

We suggest that this problem is intensified by contemporary RT-related work which, in its strive to supplant the need for human contact with extensive and traceable project histories, rationales, decision records, and the like, does not prepare appropriate foundations to actively assist the identification of RE participants. Pursuit of this objective disregards a fundamental working practice which we have found most likely to underlie the continued citation of RT problems.

In some cases, this entire problem of identifying RE participants was found to be due to organisational or project politics, which actively prohibited any contact with the original sources of requirements. We also found that, in those projects where documents are literally "thrown over the fence", it was not uncommon for access to those involved earlier on to be forbidden in later phases. These findings imply that many of the reasons for this problem are political in nature so could really only be addressed by re-examining the organisational policies of those projects in which they are experienced. We return to this important issue later on in the thesis.

6.3.2. Consequences

The inability to locate and access those involved in the production and refinement of requirements was, not only the most commonly cited problem across all the practitioners in our empirical investigations, but further reported to be a direct and major contributor to the following problems also experienced:

(1) The inability to prioritise requirements and so plan for development work. This was found to be almost impossible without access to the original sources and other original stakeholders.

(2) Poor collaboration and coordination in RE. The invisibility of changing work structures and changing responsibilities was found to make it difficult to transfer information amongst
appropriate parties, to integrate work, and to assign work to those with relevant experience.

(3) An out of date RS. It was found that an RS either evolves poorly, or not at all, when those originally responsible for its production are not involved in its evolution.

(4) The inability to re-evaluate and refine existing work in a controlled manner. This was found to be due to the inability to re-access the context in which work was originally carried out, especially the original stakeholders.

(5) Unproductive conflict resolution, decision making, and negotiation. This was found to be because the tools which are used to support these activities do not address the problem of locating those who should participate in them in the first place.

(6) Slow realisation of requirements change. This was found to be because the most time-consuming aspect is often the identification of all those to involve in the change process, particularly those who need to initially evaluate the change proposal.

(7) Progressive deterioration as a result of requirements change. This was found to be because it is common to incorrectly identify or overlook all those that need to be informed of any change.

(8) Difficulties in dealing with those individuals who leave a project and with integrating new individuals. This was found to be because some requirements often disappear, and new requirements often materialise, with participants. Also, because individual work responsibilities need to be pin-pointed if they are to be successfully re-assigned to others.

(9) Little reuse of requirements, else disaster when requirements are reused. This was found to be because reuse of requirements is mainly practiced successfully when those responsible for their original production are either directly involved or readily accessible.

6.3.3. Confirmation

The importance of the ability to identify RE participants is hinted at in some other work. A study at the U.S. DoD reported that: "The source of the requirement oftentimes is not its originator, but may be a intermediary who makes the interpretation, and therefore provides another twist to the requirements. It is important to capture information about this intermediary to help identify the history of requirements" [Harrington & Rondeau 1993]. A study of the software design process for large projects stressed that: "Large projects required extensive communication that was not reduced by documentation. Project staff found the dialectic process crucial for clarifying issues. Particularly during early phases, teams spent considerable time defining terms, coordinating representational conventions, and creating channels for the flow of information" [Curtis et al.
1988]. Furthermore, in a comprehensive analysis of the cause of software errors, recommendations were made for modularising responsibility and promoting informal communication [Lutz 1993].

Our investigations independently consolidate such findings and recommendations. They make it apparent that RT problems will continue to be evident where responsibility for project information, particularly RE information, cannot be accurately identified throughout the duration of a project, and so where appropriate individuals and groups, or suitable alternatives, cannot be located for required informal communication.

6.4. Lessons from Practice

Before examining how we could assist this more fundamental problem, it is important to note that we found certain project characteristics which actively encouraged the occurrence of this problem in our empirical studies. In projects consisting of many individuals split into a number of teams, the ability to locate and access the human sources of requirements information was found to be either impossible, time consuming, or unreliable. The characteristics of those projects which led to this problem are listed below. Interestingly, these characteristics were amongst those identified elsewhere in our investigations as high contributors to project failure.

(1) A lack of shared or project-wide commitment, with no visibility of ownership and a lack of accountability, with the phrase "not invented here" being very common.

(2) Little cross involvement in work, and localised views of information, thus making it difficult to pin down the overall state of work or knowledge.

(3) Poor communication and distribution of information amongst teams, leading to much information loss, as well as the development of cliques over time.

(4) Changing notions of ownership and responsibility, due to continually changing work structures, and due to the turn-over of team members.

In contrast we found that, in projects consisting of few individuals, the ability to locate and access the human sources of requirements information was not so much of a problem. Where there were no problems, this was attributed to the project characteristics listed below. This time it is worth mentioning that these characteristics were also identified as high contributors to project success.

(1) Clear visibility of participant responsibilities and knowledge areas.

(2) Clarity of working structures and working relations.

(3) Individuals who acted as common threads of involvement throughout the project and across
project boundaries.

(4) A strong sense of team commitment, accompanied by joint ownership.

To help the rapid identification of appropriate individuals or groups, there are obvious benefits to be gained from ensuring that those project characteristics which assist the ability to keep track of all contributors and their contributions are more widely experienced in all projects.

6.5. How can we Tackle the Crux of the RT Problem?

More information needs to be provided about the social structure of the RE process. This can be given by augmenting any information that is obtained in the RE process with details of those individuals and groups who have contributed. Such details must be updated to reflect the evolving and changing nature of those who contribute as changes are made to this information itself over time. This implies the need for a model of contributors and contributions. With such a model, pre-RT would provide the ability to trace those involved in requirements production and refinement.

We suggest that the ability to trace RE participants is likely to have a positive impact on quality. This is because this facility could be used to account for the fact that quality is socially defined, socially evaluated, and only accepted within a social setting. Although the requirements which drive quality may be well defined, and even imported from agreed standards, what is specified will often mean different things to each individual involved: "probably the most common error made in quality assurance is the assumption that there is a common understanding of what 'high quality' software actually means. No such common understanding exists. Situations arise where different software engineers strive, in a mutually antagonistic way, to ensure that particular, but different, product attributes are achieved" [Sommerville 1989]. Access to those defining or importing the requirements being used to drive quality development is often the only way to ensure a shared interpretation and to assess the results.

7. Supporting Work

We are unaware of other research explicitly directed at the above issue of providing a way to trace RE participants. However, our work described in the remainder of this thesis has been influenced by a number of disciplines, ranging from process modelling through to work in the social sciences. In this section, we summarise some of these areas and discuss ideas they have to offer.
7.1. Process Modelling

Curtis and colleagues provide a thorough overview of process modelling [Curtis et al. 1992]. They point to the use of such modelling to represent the processes through which work gets done and explain how it can be applied in business process re-engineering, coordination technology, and process-driven software development environments. Rather than replicate this overview, we briefly mention how work modelling systems and software processes, and organisational modelling in particular, is of interest when examining how best to model those participating in the RE process.

7.1.1. System and Software Process Modelling

Concern for systems and software process modelling originated from a recognition that the early models of the systems and software lifecycle, like Royce’s waterfall model [Royce 1970 (cited in [Davis 1990])] and Boehm’s spiral model [Boehm 1986], were primarily managerial tools and ineffective for representing actual development practices. Although today’s process models still provide abstract descriptions of the activities by which systems and software get developed, they increasingly do so in a manner which is more representative, finer-grained, and further provides scope for process guidance and coordination [Mi & Scacchi 1992, Nuseibeh 1994]. The proliferation and diversity of today’s process models, along with tools which support these [Garg & Jazayeri 1994], is a reflection of the numerous ways in which different phases of systems and software development work can be viewed. For instance, as a:


(2) Contractual process [Dell 1986, Dowson 1987, Fikes 1982].


(8) Social action process [Hirschheim et al. 1991].

The most recent process models provide various details about the nature of the relationships between agents, their activities, and their products, of which the RE process meta-model of the
Part II: Problem

previously mentioned NATURE project is a particular example [op. cit.]. However, such models are not explicitly designed to model participation in RE, to keep track of changing patterns of participation, or to further use this information for reasoning about RT issues. Efforts to increase the amount of RE process knowledge tends to touch upon all concerns rather thinly, whereas we have argued the need to focus in detail on the knowledge of those individuals and groups involved.

7.1.2. Organisational Modelling

Within the RE community there has been much recent interest in modelling the organisational environment in which a system or piece of software is to operate. This activity is also frequently referred to as enterprise modelling [RESG 1995]. The purpose of such modelling is to ground the development process in an understanding of the organisation concerned and so obtain organisational requirements. It must be noted that this activity has been an integral aspect in the automation of organisational processes for many years and is nowadays supported by an increasing number of tools [Spurr et al. 1993]. The main problem confronting such efforts to obtain an accurate model of an organisation is that any organisation can be characterised in many ways and analysed from numerous perspectives [Jirotka et al. 1992, Litterer 1963, Morgan 1986, Morgan et al. 1983, Pugh et al. 1971, Silverman 1970]. Each model places different concepts at the forefront and so only provides a partial view of an organisation. Necessarily, the forms of organisational modelling now prevalent in systems and software development work specifically select those entities and relations within an organisation to focus on dependent on those restricted aspects of an organisation under scrutiny. For instance:

(1) To capture the goal structure of an organisation, models can be constructed which represent the various organisational goals, their interrelations, their decomposition into business rules, work plans, and so forth, such as the enterprise model of the F³ project [Bubenko et al. 1994].

(2) To capture the workflow of an organisation, models can be constructed which represent the flow of data or resources between agents, like the Information Control Net model of [Ellis 1979, Ellis & Nutt 1980].

(3) To capture the intentional structure of an organisation, models can be constructed which represent various dependencies amongst agents, such as their inter-dependencies with respect to resources, tasks, and goals [Yu 1993, Yu 1994].

(4) To capture the patterns of authority and responsibility operating within an organisation, models can be constructed which represent the authority and responsibility relations between agents and tasks [Sushil & Raghunathan 1994].

(5) To capture the role structure of an organisation, models can be constructed which represent
the responsibilities and obligations between interrelated work roles, as in the ORDIT method [Blyth et al. 1993, Dobson & Strens 1994].

To capture details about the human activity system operating in an organisation, models can be constructed which represent the activities or interaction that takes place between agents, either by describing the communicative action that takes place [Auramaki et al. 1988], or by describing how they are mediated by language, tools, artifacts, and so forth [Bødker 1989].

What is interesting is that the above mentioned forms of organisational modelling have not yet migrated to studying the moment-by-moment working organisation of those involved in the systems and software development process itself, a classic example of a dynamic open system in a changing environment [Lievegoed 1973]. Current forms of modelling tend to be used to provide static, formal, inflexible, and rationalised descriptions of aspects of an organisation, whereas more dynamic variants of the above models are needed to account for changing and spontaneous work structures, work roles, and interrelations. Such issues have received a high profile for many years within the office automation literature [Gasser 1986, Gerson & Star 1986, Hewitt 1986] and the information systems research is continually looking into ways to model organisations as dynamic systems so as to gain insight into their more informal structures and processes [Sol & Crosslin 1992]. For the purpose of this thesis, we are only concerned with constructing a model to represent the evolving pattern of relations between requirements artifacts and those who participate in their production and refinement, but we can find no dedicated model to do this in the literature.

7.2. Project Management Techniques and Tools

There is plenty of project management software which functions to assist the above forms of modelling [Badini & Whitehouse 1989]. These commonly support the traditional project management activities of planning, scheduling, and allocating resources. Many commercial CASE tools increasingly incorporate facilities to represent formal organisational charts, timelines, and work breakdown structures, and so provide the opportunity for activities like workflow analysis. However, they typically model formal and static organisational structures, and predefined work plans which have been decomposed from an agreed RS. Notions such as roles, authority, and responsibility are also generally fixed at the onset of a project. They hence tend to presuppose deterministic organisations and working practices. The drift between what is modelled, what actually took place, and what is the case in later project life, can often be dramatic. Even so, there are many facilities provided by such tools which could be exploited in conjunction with RT to provide added value, like impact analysis in conjunction with the projected cost of such change, plus its further implications for the budget and critical path.

Of particular interest amongst all such project management techniques are responsibility tables,
matrices, or charts. As illustrated in Figure 7, these are used to assign people to project tasks, so can provide some interface between personnel and their actions. They are used mainly for project planning and monitoring. They record relations like: (a) what parts of a project people are responsible for; (b) what parts of a project people need to approve; or (c) what parts of a project people need to be consulted or informed about. However, such project management techniques and tools will not be suited to tracing changing patterns of personnel involvement in the RE process until they account for and reflect the emergent properties of organisational structures and the dynamics of working practices.

![Figure 7: A basic responsibility matrix used in project management.](image)

**7.3. Social Sciences**

We have been influenced by much work which could be said to fall under the broad heading of the social sciences since it provides some insight as to the kinds of issue to consider when attempting to model the social structure underlying the RE process. Most of this relevant work is referenced at appropriate places in Part III of this thesis. In particular, there is an abundance of work which investigates how facts and artifacts are both related to, and influenced by, the social structures from which they arose [Barnes & Edge 1982, Bijker et al. 1987, Callon et al. 1986, Latour 1987, Latour & Woolgar 1979, Law 1991, Law 1994, Pickering 1992]. Another relevant area is the use of social theory to provide a better understanding of software and systems development and use in general [Hirschheim & Klein 1989, Kling 1980, Murray & Woolgar 1991], and of requirements development and use in particular [Jirotka & Goguen 1994]. A further relevant area is work examining the citation networks which exist across papers and books [Crane 1972, Gatrell 1984].
This is because they can be used to reflect the networks of communication that exist amongst people and can be used to uncover implicit information such as the structure and immediacy of influence amongst people. A simple citation network can be derived from the matrix given in Figure 8 showing the connectivity between a collection of papers.

![Figure 8: A basic matrix used to form citation networks.](image)

7.4. Document Studies

We have also been influenced by much work which we place under the broad heading of document studies. Again, most of the relevant work here is referenced at appropriate places in Part III of this thesis, and includes the consideration of issues like: (a) how authorship gets attributed to documents; (b) how works which have been achieved through collaboration come to be attributed to a single author; (c) the different ways in which an author label functions; (d) the various strategies that can be employed to authenticate authorship and determine authorial styles; and (e) the very different issue of document ownership [Barthes 1979, Biriotti & Miller 1993, Chartier 1992, Foucault 1979, Harari 1979, Rose 1993, Stillinger 1991]. All such issues are essential to clarify in order to deal with further issues such as how notions like authorship and authority are diffused as information in documents is dispersed and proliferated. Of additional interest here is work which reproblematises the above notions and examines how they are socially constructed, like that seen in [Bonnycastle 1991].

It has been useful to examine work on the nature of record-keeping, policies for introducing record-keeping practices, and the characteristics and use of written records. For instance, Raffel
Part II: Problem

explains how records are not simply records or conveyors of information, but also resources in the organisation of conduct, so goes on to describe the roles they play in mediating communication and activity [Raffel 1979]. A record is therefore not merely something which is itself accessible, but is something which makes other things accessible. Wheeler notes how one thing they serve to make accessible are the people involved in the recorded practices: "... putting matters on record is a means of making visible, and hence accountable, the activities of organisations and their members" [Wheeler 1969]. This is important because Wheeler maintains that the truth of records can be indirectly assessed by the reliability of the record-keepers and the knowledge of the relations between the other personnel involved. It is this wider role of requirements information that we need to exploit in RE documentary practices.

7.5. Role Theory

As suggested in Box 10, the notion of "role" is relevant because it can provide a handle for examining the ways in which individuals and groups are involved in the RE process. Roles can also provide the means through which we can look into particular aspects of this process like: (a) lines of command; (b) responsibilities; (c) authorities; (d) job specifications; (e) obligations; (f) social bargains underlying cooperation; and so on. For our purposes, we simply consider a role to be a cluster of rights and obligations, the various perspectives in role theory being covered elsewhere [Banton 1965, Biddle & Thomas 1966, Emmet 1966, Jackson 1972, Linton 1936, Ruddock 1969].

"Role, then, is the basic unit of socialization. It is through roles that tasks in society are allocated and arrangements made to enforce their performance ... all behaviour must originate from a person in some position or other in relation to other people ... all individual behaviour which is in any way related to other individuals can be seen as role behaviour."

Box 10: The notions of "role" and "role relation" are central to the analysis of social situations (taken from [Ruddock 1969]).

Those involved in the RE process assume both relational and non-relational roles. Together, these define their expected behaviour with respect to the work itself and to the others involved. The role modelling commonly used within the domain of systems and software engineering mainly encompasses subsets of the achieved roles shown in Table 8. Typically, the specification of non-relational roles does not evolve as the attributes which define them develop in practice. Also, the specification of relational roles does not capture those which are informally formed by the actual working relations entered into. Therefore, role modelling is generally used here to declare static job descriptions and predefined working relationships.


<table>
<thead>
<tr>
<th></th>
<th>Ascribed role</th>
<th>Achieved role</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Non-relational role</strong></td>
<td>e.g., age.</td>
<td>((a)) Proprietary</td>
</tr>
<tr>
<td></td>
<td></td>
<td>((b)) Expressive</td>
</tr>
<tr>
<td></td>
<td></td>
<td>((c)) Service</td>
</tr>
<tr>
<td><strong>Relational role</strong></td>
<td>e.g., kinship.</td>
<td>((a)) Symmetrical</td>
</tr>
<tr>
<td></td>
<td></td>
<td>((b)) Asymmetrical</td>
</tr>
</tbody>
</table>

**Table 8:** A partitioning of role types (according to [Nadel 1957]).

If we further consider Banton’s role tree shown in Figure 9, we can see that the current use of role modelling in systems and software engineering really only accounts for the branches of this tree, though very occasionally the trunk. However, it is the leaves of this tree which are able to tell us about the minutiae of working practices. Transient roles are contingent notions which are dynamically assigned and emerge in interaction. They capture notions like the arbitrators and spokespeople for periodic aspects of the process. They make detailed analysis of actual working arrangements possible as they enable us to study how people are simultaneously involved in many intertwined processes and role relationships. They further provide the foundations for examining how roles shift and role systems evolve. These are the roles with which we need to be most concerned if we are to understand the social structure underlying the RE process but, to date, there has been little research done in this area.

**Figure 9:** Banton’s role tree (adapted from [Banton 1965]).
8. Summary of Part II

In Part II, we have analysed the multifaceted nature of the so-called "RT problem" that many practitioners refer to, and have explained why there is unlikely to be an all-encompassing solution. We have outlined the method we used for collecting data about the problem and, from this data, we have shown how the problem itself can be decomposed into a number of more fundamental and underlying problems. We have further shown how these problems stem from conflicting and changing viewpoints regarding what RT is, what the problems with it are, and what it is meant to achieve. Through the provision of a conceptual RT framework, we have introduced a distinction between pre-RT and post-RT, and further described their current support and long-term potential for alleviating RT problems. We have argued that, in order to achieve any order of magnitude improvement with RT problems in the longer-term, there is a need to re-focus research efforts on addressing the issues of pre-RT and so provide a more comprehensive approach to RT. We have discussed the problems confronting, the requirements for attaining, and ways in which progress could be made with pre-RT. Although the obvious direction appears to be towards gathering extensive requirements-related information and making this traceable, based on empirical evidence we have argued that these advances will still need to be augmented with the ability to identify those individuals and groups in a position to supplement it, since the ability to trace those who have contributed in RE was found to be a fundamental working practice. This practice provides what is often the only way to explain and assess change, reason about real needs, deal with the inevitable shortcomings of information which is documented, and so abate RT problems. To tie the appropriate people into the RT equation to address this central issue, we have suggested that the social structure in which requirements are produced, specified, maintained, and used, needs to be more significantly recognised and explicitly supported.
Part III: Approach

1. Overview of Part III

We have found the crux of the RT problem to be the difficulty in identifying the human sources of requirements, requirements-related information, and requirements-related work. In turn, this indicated that details about those individuals and groups who have contributed in the RE process is crucial information to make traceable. In Part III, we describe an approach which makes the social structure underlying RE explicit. This approach is based on modelling the contribution structure underlying requirements artifacts. We first mention the kinds of question which are currently problematic to answer in practice and lead to claims of RT problems, the reasons for this difficulty, and so describe the setting into which the approach is to fit. We indicate the scope of the approach, list the fundamental requirements that it needs to meet, clarify some of the terminology and assumptions it makes, and provide an outline of the steps it involves. In further describing each of the steps, we: (a) describe a scheme to relate agents and artifacts, to define rudimentary contribution structures, which delineates the capacities in which agents can contribute; (b) show how the information this scheme captures can be used to infer details about the social roles and commitments of agents, and so develop contribution structures; and (c) describe a categorisation for artifact-based RT relations, to illustrate how they impinge on the definition of contribution structures. We then clarify how the approach addresses the crux of the RT problem using modelling techniques and examples. In particular, we show how the problematic questions can be addressed, and thereby show how dedicated forms of personnel-based RT can be supported in conjunction with the more conventional forms of artifact-based RT.

2. Revealing Participants in Requirements Engineering

In this section, we summarise why there is a need to capture relevant information that can be used to model the social structure underlying the RE process. In this thesis, we use the term "social structure" to refer to the overall system of agents participating in the RE process, along with the various relationships they are involved in. A description of what is generally meant by the term "social structure" is given in Box 11, whilst more detailed treatment can be found in [Olsen 1968].
We first provide a list of questions that are frequently asked about those individuals and groups involved in the RE process, then describe the deficiencies with prevailing practice which make informed traces of those involved unworkable, so these questions difficult to answer. Finally, we mention some of the political issues which are likely to hinder the acceptance of any proposed changes to practice.

"... the variety of institutions and organisations, and the relationships between them, together with the inequality between individuals and their interrelationships, constitute what is called the 'social structure' of a society."

**Box 11: A definition of the term "social structure" (taken from [Burns 1993]).**

2.1. Frequently Asked Questions

The primary objective of modelling the social structure underlying the RE process is to make it possible to trace the human sources of requirements, requirements-related information, and requirements-related work. This is to enable practitioners to answer those questions which, if left unanswered, were found to result in their citation of RT problems. Examples of such questions include:

1. "Who has been involved in the production of this requirement and how?"
2. "Who was originally responsible for this requirement, who is currently responsible for it, and at what points in its life has this responsibility changed hands?"
3. "At what points in this requirement's life have the working arrangements of all those involved been changed?"
4. "Who needs to be involved in, or informed of, any changes proposed to this requirement?"
5. "What are the ramifications, with respect to the loss of possible requirements-related knowledge, if a specific individual or group leaves a project?"
6. "To whom should I refer for more information about this requirement?"
7. "Within the remit of which group do decisions about this requirement lie?"
8. "Who was responsible for copying this piece of information into this requirement?"
9. "Who uses this requirement and how?"
10. "On which requirements do certain agents collaborate together and how?"
2.2. Practical Problems

We found a number of problems relating to how details of those who have participated in the RE process are predominantly kept in practice. Where not absent, such details were generally found to be inadequately described and maintained, and also not given in a very useful form. This situation was mainly a consequence of the coarse-grain and snapshot nature of the labels used to denote participation.

2.2.1. Coarse Labels

We found that typical records of participants in the RE process usually consisted of a list of names in either an "author" or "owner" field of the documents produced. Simply appending such labels to a document results in relatively coarse notions of authorship or ownership. Moreover, these notions are conventionally used to refer to those individuals and groups who wrote the documentation, as opposed to those who inspired or formulated the content therein. They neither account for those situations in which, say, the individuals in the group collaborated or acted independently on various fragments of the document, nor do they account for the nature, scope, and degree of their contribution. Where the product of a collective is attributed to an individual, where the product of an individual is attributed to a collective, or where the different roles of all those contributing is not demarcated, it is hard to identify who to call upon when different types of problem or change request are subsequently encountered. This means that later problems or questions often get answered by those who are not necessarily best placed to do so. The typical situation arising from such coarse-grained labelling is illustrated in Figure 10. This figure further illustrates why questions regarding the origin of a requirement, or those involved in its production process, can only be handled in a rather simplistic manner at present. It suggests that participation details need to indicate the status of those agents who are party to the production of requirements artifacts, along with the mode of their participation.


Figure 10: Typical coarse-grained labelling conventions. Artifacts produced in RE tend to be labelled with the name of an author, where this is used to refer to the individual or group responsible for producing the actual artifact. Little regard is given to recording details about those individuals and groups who were actually involved in generating the information in the artifact. This is why the end products of RE tend to lose traces of the original contributors or owners of the information, and why it is usually only possible to identify those who wrote the information down along the way.

2.2.2. Snapshot Labels

We also found that the current way of recording details about the participants in the RE process does not provide a suitable structure in which to represent any changing patterns of participation as the document contents evolve and are used elsewhere. We found that those documents which had been changed were generally characterised by the addition of further names in the "author" or "owner" field, by appended notes, or by the addition of official change request forms, as illustrated in Figure 11. This practice was found to compound RT problems, especially as the size and longevity of a project increased. The coarse-grain labels do not delimit who has altered parts of a document. The annotations are frequently not integrated with the original document content, so are inaccessible. Such additional records can soon become too unstructured and unwieldy for analysis purposes. A repercussion which effects quality is that agent commitment to developing artifacts, as well as to each other, becomes fragmented and lost over time. This suggests that participation details need to be evolvable and open to finer delineation over time.
2.3. Political Problems

If we are to bind individuals and groups to the artifacts they have produced in the RE process, there is a need to de-politicise the issues involved if this is to be organisationally acceptable. This is because, to be observable to others is to be accountable to them. This further introduces notions like commitment, responsibility, accountability, and so forth. There are obviously both positive and negative ways to interpret and use such notions. Where such notions are used to attribute blame, and even to effect punishment, this can only lead to a culture of secrecy and rivalry. Where such notions are used to reward, or to identify problem areas early on and so respond rapidly to them, this can lead to a culture of visibility and participation. Although it has long been noted that a poor accountability structure exacerbates many project problems [Boehm 1976], the majority of today's work settings still actively promote practices which obscure such notions, largely because they are all too often used in the negative manner described above. Another reason compounding this is that the required change in culture that is necessary to use such notions in a positive manner

Figure 11: How changing authorship details are typically accounted for. As the internal content of artifacts is changed over time, it can become increasingly difficult to partition the separate areas that different individuals and groups have authored. This is largely due to the snapshot labelling of authors and owners and the various attempts to get around this as these notions fragment with time.
must flow from top management and down through an entire organisation [Brady & DeMarco 1994].

Many others have also argued that, where such notions are used positively, benefits will be reaped in practice. For instance, Nissenbaum describes how the value of accountability is currently undermined in systems and software development, then goes on to argue that accountability is one of the most powerful tools that we currently have for bringing about better practices and for consequently producing better systems: "A community that insists on accountability in which agents are expected to answer for their work, signals esteem for high-quality work, and encourages diligent, responsible practices ... the call for accountability remains a standard worth restoring, and one whose achievement would be a source of professional pride" [Nissenbaum 1994]. The ability for practitioners to take responsibility, and so be held accountable for their own contributions, has even more recently been declared as a fundamental principle of software engineering: "There are no excuses. If you develop a system, it is your responsibility to do it right. Take that responsibility. Do it right, or don't do it all" [Davis 1994]. Furthermore, it has frequently been argued in numerous other disciplines how notions such as accountability, commitment, and ownership are crucial for managing teams [Kahn & Kram 1994, Morgan 1990, Walton 1992].

3. An Approach Based on Contribution Structures

To provide the ability to trace detailed information about the social structure underlying the RE process, we need to provide a clearer picture of the overall system of individuals and groups involved in the process, along with the various relations they are involved in. In this section, we confine ourselves to a manageable portion of this problem sufficient to argue our thesis, then outline the fundamental requirements for an approach to address it. The terminology used, the assumptions made, and the rationale behind these requirements follow. For orientation purposes, we provide an overview of the approach, before it is described in more detail in the following sections.

3.1. Scope

Through our empirical work, we found that practitioners predominantly claim to have experienced RT problems when, being unable to retrieve the particular information about requirements production and refinement that they want from a project repository, they have further been unable to identify those who would be in a position to supply it. This situation was found to be caused by the way in which details of participation are currently recorded and maintained, and was no doubt a reflection of the absence of recommended guidelines to achieve this in the recognised RE standards. This means that we limit our scope to the traceability of information relating to the tangible artifacts produced in the RE process. To be more specific, we are particularly concerned
with that information which illuminates the social structure underlying these early artifacts, though we acknowledge the long-term desirability to be able to trace all the participants involved throughout the systems and software engineering lifecycle. Now, as this social structure is defined through all the activities that individuals and groups are engaged in throughout the RE process, it would be extremely difficult to obtain with complete accuracy. For this reason, we necessarily restrict our scope to capturing information about the direct contributors to the tangible artifacts, and so reflect what we instead called the "contribution structure" underlying the RE process. We believe this will provide a reasonable enough approximation to the true social structure for our purposes, bound a potentially vast structure in a practical way, and also focus more closely on the kinds of participant information that practitioners are most likely to want.

3.2. Fundamental Requirements

The fundamental objective is to build and maintain a dynamic model of those agents who have directly participated in the production of tangible requirements artifacts which also provides the potential for personnel-based RT. The basic requirements for such a model are:

(1) A way to differentiate the various ways in which agents have contributed to requirements artifacts, to hence distinguish the status and mode of their contribution, which can also supply the building blocks with which to model contribution structures. This means that any scheme chosen to characterise the relation between agents and artifacts needs to make it possible to build progressively intricate contribution structures from relatively simple ones, and in such a way that their properties can be determined through an analysis of their building blocks.

(2) A way to account for the numerous relations that exist within and between the requirements artifacts themselves, as these will have an impact on the exact nature of agent contributions, and so would provide further information about contribution structures. This means a need to supply a way in which agents and artifacts can co-evolve to prevent contribution structures from remaining static and becoming outdated.

(3) A suitable basis for reasoning with and about the information modelled by contribution structures. This would be needed to handle growing numbers of artifacts, agents, and interrelations. It would also be needed to enable selective information about contributing agents to be retrieved and to thereby extend artifact-based RT with sophisticated personnel-based RT queries.

The requirements we list above mean that a dedicated approach is needed to guide the definition, maintenance, and use of this model. Such an approach must require minimal practitioner effort for organisational acceptance. It must either automatically capture and maintain as much of the
Part III: Approach

information as possible, on-line and as a by-product of working practices, else more incentives must be provided to do this manually.

3.3. Terminology and Assumptions

Here, we mention some of the assumptions upon which the approach is based, and further explain some of the terminology which we use in its description.

3.3.1. Why RE Artifacts and Agents?

We focus on the tangible artifacts produced in the RE process, along with the agents who have directly contributed to their production, for four main reasons:

(1) A deliverable-driven development culture tends to dominate in industrial settings, so tangible artifacts are often readily available.

(2) It is through the artifacts that are produced that the links between agents become defined and reinforced.

(3) Artifacts mediate the subsequent activities that agents engage in.

(4) Artifacts hold with them an explicit notion of commitment which is something we want to exploit later.

3.3.2. Artifacts (Contributions)

By the term "artifact", we refer to any communicative occurrence in the RE process with a physical existence of its own. Thus we include within the scope of our definition informal email notes and diagrams, through to formal change request forms and requirements documents. Conversations, meetings, gestures, handshakes, and similar, would only be considered artifacts by our definition if they were explicitly recorded in a manner which gave them physical identity and persistence. We place no restriction on the information that can be represented in an artifact, the media this can be represented in, or the level of granularity at which contributions can be delineated. In fact, we are not concerned with the actual internal content of artifacts themselves for now, just the fact that artifacts can either be:

(1) **Primitive**, by which we mean composed of no other artifacts.

(2) **Composite**, by which we mean composed of other artifacts.

We assume that all the tangible artifacts that are generated and exchanged in the RE process, whether intermediate or final documents, are held in an on-line artifact repository which handles
the types of artifact-based RT that we suggest later in Section 6. We make this assumption due to reasons such as: (a) the increasing maturity of digital imaging technology and optical character recognition [Reinhardt 1994, Seachrist 1995]; (b) the sophistication of the techniques and applications available for document management [CGSA 1995a, CGSA 1995b, Dewire 1994]; and (c) the ability to configure existing tools to provide the artifact-based RT we find appropriate. Although the availability of both public and private tangible artifacts is desirable for a comprehensive approach, this is likely to be subject to political problems, so something best left up to the policy makers to decide.

3.3.3. Agents (Contributors)

By the term "agent", we refer to the human participants in the RE process. As it is possible to attribute artifact production to specific individuals or to groups of individuals, we distinguish between these two types of agent. We assume that the various agent details that are required, such as their identities, positions, authority, rights, and duties, are held in an organisational repository of some form. Relational information could also be held about agents to model diverse forms of predefined formal organisational structure or role structure, though we are not concerned with such aspects for now.

3.3.4. Contribution Relations

We use the term "contribution relation" to mean any physical relationship that exists between an agent and an artifact as a consequence of the former contributing to the latter's production in some way. This is inherently a bi-directional relationship as an agent contributes to an artifact and an artifact is contributed by an agent. The end points of this relationship are assumed user-definable, so can join agents and artifacts at any level of granularity. This means that contributions can be of any size, nested, overlapping, embedded, and so forth. This also means that contributors can range from isolated individuals to intricate webs of individuals. The impact of these differences in granularity are illustrated in Figures 12 and 13. The precise nature of contribution relations is the subject of Section 4.
Figure 12: A coarse-grained contribution relation. Such a relationship between a group of agents and a composite artifact does not serve to differentiate the individual contributions.

Figure 13: Finer-grained contribution relations. Such relationships between a group of agents and a composite artifact serve to differentiate the individual contributions.

3.3.5. Contribution Structure

The contribution structure underlying a requirements artifact refers to all the contribution relations that have been defined for it. Obviously, the potential richness with which this contribution structure can be described depends upon the way in which the contribution relation between agents and artifacts is defined. As artifacts often depend upon the existence of other artifacts in some way or another, or are themselves decomposed into component artifacts, this description also
depends upon how well these artifact-based relationships are defined and taken into account. By extension, the contribution structure underlying the RE process is therefore described in terms of the contribution structures of all the tangible requirements artifacts it produces and their interrelations. An abstract depiction of this is given in Figure 14.

![Figure 14: The contribution structure underlying RE. This refers to all the defined contribution relations for the tangible artifacts produced, along with all their interrelations.]

### 3.3.6. Artifact-Based RT and Personnel-Based RT

Throughout this thesis, we make a distinction between artifact-based RT and personnel-based RT. This distinction is illustrated in Figure 15. The former makes use of the relations which exist between requirements artifacts, often defined at varying levels of granularity, and is what conventional notions of RT provide. The latter makes use of the relations which exist between artifacts and agents, namely the contribution relations, to further determine the relations which exist between those directly involved in the RE process as a consequence. It must be noted that this does not necessarily reflect the organisational and working relations that can be determined from the typical types of information held in organisational repositories and work charts. Together, artifact-based RT and personnel-based RT, accounting for both pre-RT and post-RT, would provide a comprehensive approach to RT which would be able to address the outstanding
RT problems we have identified.

![Diagram of RT problems with agents and artifacts]

**Figure 15:** Artifact-based RT and personnel-based RT.

### 3.3.7. Logical Basis

As mentioned in Section 3.2, it would be desirable to provide the approach with a suitable basis to coordinate model building, and to further establish rules for the use and development of this model. In forming a relation of contribution between an agent and an artifact, a public engagement is made which restricts freedom of action, and also introduces the notion of "commitment" as used by [Finkelstein & Fuks 1988]. Box 12 further defines the conditions through which such behavioural commitment is generally established. We therefore suggest that the model of the contribution structure could ultimately be defined in terms of agent commitments. By representing these commitments in logic, we could provide rules to: (a) maintain an up-to-date model of the contribution structure; (b) recompute the contribution structure following requirements change; (c) handle the sorts of reasoning required to selectively trace RE participants; and so forth. Here, it may be possible to make use of some of the work in the distributed artificial intelligence literature seeking to develop a computational theory of commitment [Avouris & Gasser 1992, Bond 1990, Bond & Gasser 1988, Gasser 1991, Gasser & Huhns 1989, Hewitt 1991, Huhns 1987, Jennings 1993].

123
"Commitment comes about when an individual is bound to his acts. Though the word bound is somewhat clumsy, what we mean by it is that the individual has identified himself with a particular behavior. Three characteristics bind an individual to his acts and hence commit him. They are the visibility, the irrevocability, and the volitionality of the behavior. By manipulating these three characteristics, an individual can be made to be more or less committed to his acts and their implications."

**Box 12:** How behaviour binds an individual to an organisation (taken from [Salancik 1977]). Note that this can refer to how an act of an individual in the RE process can bind this individual to the organisation conducting the process.

### 3.4. Steps of the Approach

For reader orientation, we outline the main steps of the approach, and illustrate them in Figure 16. The details of the approach are then given in Sections 4 through to 6.

Modelling the contribution structure underlying an artifact involves augmenting the artifact with details of those agents who have contributed in the different capacities suggested by the scheme we describe later in Section 4. It then involves manipulating these details to ascertain information about social roles and agent commitments, which we describe later in Section 5. This would model a rather simple and static model of the contribution structure, so only provide the basis for rudimentary forms of personnel-based RT. This is because it does not account for the relations within the artifact itself and between other artifacts, so the contribution structure cannot evolve as the artifact itself evolves and becomes tied into a more intricate web of artifacts. Modelling a more intricate and dynamic contribution structure therefore depends on also taking the artifact-based RT relations into account when describing the nature of contributions, which we describe later in Section 6. These enhancements are crucial because the detail with which the contribution structure can be described directly influences how selective the personnel-based RT can become.
4. Relating Agents and Artifacts

In this section, we describe our scheme for relating agents and artifacts, and explain the limitations of using a singular relation labelled "contribution". In introducing our scheme to do this, we provide an account of the work upon which it is based, and then describe the three elementary types of contribution relation it proposes. We further show how these relations could be qualified to provide more information about the nature of these different types of contribution. Finally, we give an example to show how these relations can be used to define rudimentary contribution structures.
4.1. Representing Contribution Relations

Although the relation between agents and artifacts could be defined using terms like "contributed_to" and "contributed_by", these would not distinguish different types and degrees of contribution, so would not meet the basic requirements of Section 3.2. As shown in Figure 17, they would lead to flat, coarse, and network-like models of the contribution structure. As this would provide little real potential to reason about RE participants, there is evidently a need for some finer discrimination in order to impose more structure. The kernel of the approach therefore lies in providing a suitable scheme with which to describe this contribution relation. It must reflect the fact that many agents may be party to the production of an artifact, so differentiate the nature of each of their contributions, and must also provide a suitable basis for modelling a progressively more granular and layered contribution structure.

![Figure 17: A flat and coarse model of the contribution structure.](image)

4.2. Foundations of a Scheme

The scheme we propose to differentiate the nature of contributions is derived from work in the area of sociolinguistics and, in particular, from descriptive models of the interaction between language and social life. The reader is directed to [Hymes 1972a] for an introduction to this area. Such models aim to provide finer-grained schemes through which to describe and analyse the components of communicative situations than those schemes provided by the traditional dyadic models of communication theory, that of [Shannon & Weaver 1949] being a typical example of the latter. More specifically, our scheme is based on Goffman's work on the nature of participation...
Part III: Approach

in social encounters, which is introduced and further developed in [Goffman 1974, Goffman 1979, Goffman 1981]. We describe those aspects of this work that we are interested in and then make reference to related work.

4.2.1. Goffman's Work

Goffman's work is concerned with placing the production and reception of talk within an interactional framework. This is so that it can be studied as a component of the full physical, social, and cultural environment in which it occurred. Such a framework codifies and organises involvement and provides the means for identifying and referring to participants in an activity. However, Goffman stresses that our common-sense notions of "hearer" and "speaker" are too crude to enable such an analysis, and argues that they conceal a rather more complex differentiation of positions. For instance, the single notion of "speaker" has connotations that the agent who is physically speaking is formulating his or her own text and staking out his or her own position through it, but when reciting or reading aloud, agents can speak words they did not formulate and express beliefs they do not hold. Furthermore, they can speak for others, and hence not take the position to which these words attest. Goffman therefore decomposes these two concepts into their underlying constituents, which he refers to as participant roles, to better define the linkage between an individual and a social situation. This decomposition is fully given in Figure 18. More specifically, he refers to the set of categories obtained from the decomposition of "hearer" as the participation framework, and the set of categories obtained from the decomposition of "speaker" as the production format. The consequence of this finer discrimination is that utterances are no longer treated solely as the product of a single speaker and a single mind, but instead accounts for collaborative and multi-party settings in which more than one participant is potentially relevant. Moreover, it is suggestive of the various ways in which individuals can participate.
As we are primarily interested in modelling those agents directly involved in the production and refinement of requirements artifacts, in a manner which is amenable to additional forms of analysis and traceability, Goffman's notion of "production format" provides insight as to how the contribution relation could be represented to enable this. Goffman used this term to identify three analytical capacities in which participants can "speak", which together clarify the notion of "speaker", and are what he referred to as:

1) The **animator** of the utterance, namely the agent who is the transmitter or talking machine.

2) The **author** of the words expressed, namely the agent who composes the lines that are animated.

3) The **principal** committed to what is said, namely the agent who motivated the words or whose position they establish.

In Goffman's work, these three capacities apply to a discourse unit referred to as an "utterance event". As the incumbents of these capacities can change within the production of a single structural unit, or turn, an utterance event is the maximal unit within a structural unit in which the incumbents of the capacities are held constant. Since a single structural unit can have different individuals participating in the same capacities throughout its production, it is important to distinguish between the capacities and their incumbents.
Goffman also refers to *embedded animators, embedded authors* and *embedded principals*. These notions accommodate situations of information dependency, such as repetition and retelling, where different speakers can be embedded in current utterances. The biographical continuity this layering effect provides enables all sorts of unrestricted displacement in space and time, and with each movement closer to or further from the literal, a change in what Goffman terms "footing" occurs. This can tell us about changes in the alignment of those involved in the course of an event:

"A change in footing implies a change in the alignment we take up to ourselves and the others present ... a change in footing is another way of talking about a change in our frame of events" [Goffman 1981].

### 4.2.2. Other Related Work

Bateson's concept of "framing" is also worth noting here [Bateson 1972 (cited in [Tannen 1993a])]. Bateson used this concept to demonstrate how communicative moves can only be understood by appeal to meta-communicative messages about what is going on. In the sub-section above, we have described what could be interpreted as Goffman's particular treatment of framing within sociology. However, it must be noted that framing has also been taken up in other disciplines. Tannen describes slightly different treatments of this concept within psychology, linguistics, and anthropology [Tannen 1993b]. In the linguistic field, and in a similar vein to Goffman, Grosz and Sidner differentiate the unique roles participants maintain in a segment of conversation, and go on to show how this can be used to help understand language behaviour [Grosz & Sidner 1986]. Also in the linguistic field, Tannen describes how frames are made and how framing is made manifest in discourse [Tannen 1993a]. She further points to how the notion of "involvement strategies" can contribute to an understanding of what is going on in communication, explaining how involvement underlies all human communication, as well as how knowledge of this interpersonal involvement can facilitate understanding [Tannen 1989].

Other related ideas can be found in work examining the history of the written word. This is because such work attempts to disclose how written works are anchored in the practices and institutions of the social world [Bolter 1991, Chartier 1992]. They also examine issues like: (a) the changing principles guiding the production of texts over the years; (b) the reasons for originally constructing a notion of authorship and its changing signification over the subsequent years; and (c) the various relations that exist between texts, authors, readers, and other figures. Related ideas are also evident amidst the many strategies of structural and post-structural literary criticism [Harari 1979]. Here, the work of Foucault is particularly relevant as it examines the relationship between a text and its author in quite some detail [Foucault 1979]. Foucault explains how the notion of "author" plays a classificatory function which enables us to group texts together, define them, and so differentiate them from others. He goes on to demonstrate how there are functions that cannot be captured by this notion, like the signatory of a letter or the guarantor of a contract,
so stresses the need to clarify further the different forms this relationship between an author and a
text can take.

4.3. Contribution Format

Although Goffman's work deals specifically with talk, we extend his basic framework to deal with
the analysis of agent involvement in the production of tangible requirements artifacts. So, instead
of attributing production roles to utterance events, we attribute them to events of artifact
production. This is in recognition that requirements artifacts are produced and used within a social
environment and also that any framework depicting this environment needs a richer descriptive
structure than merely that of who the artifact "author" is if it is to be of much use. By applying
Goffman’s frame analytic method to study the social organisation of the RE process, the social
space would be partitioned according to Table 9. In this thesis, we only focus on the first quadrant
of this table. However, extending the approach to take all the quadrants and their
interdependencies into account is valuable future work, since they frequently coincide and directly
influence each other in critical ways.

<table>
<thead>
<tr>
<th>Production roles</th>
<th>Reception roles</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Participant roles</strong></td>
<td><strong>Non-participant roles</strong></td>
</tr>
<tr>
<td>Those agents directly involved in producing RE artifacts.</td>
<td>Those agents indirectly involved in producing RE artifacts.</td>
</tr>
<tr>
<td>Those agents who make use of RE artifacts for whom they have been explicitly produced.</td>
<td>Those agents who make use of RE artifacts for whom they have not been explicitly produced.</td>
</tr>
</tbody>
</table>

Table 9: Partitioning the social dimension of RE (derived from [Goffman 1979]).

For now, and in the spirit of Goffman, we use the concept of a contribution format to define the
nature of the contribution relations between agents and artifacts. Our scheme therefore delineates
three fundamental capacities in which agents can contribute to artifacts, which together clarify the
broad notion of "contribution". Shown in Figure 19, we refer to these as the principal, author, and
documentor.
4.3.1. Principal

The principal is the agent(s) whose position and/or belief is established by the information in the artifact. This is the agent who motivated the production of the artifact, is committed to what it expresses, and is responsible for its effect or consequences.

4.3.2. Author

The author is the agent(s) who put together and organised the information which is expressed in the artifact. This agent is responsible for the content and structure of the artifact.

4.3.3. Documentor

The documentor is the agent(s) who either recorded or transcribed the data which is present in the artifact. This agent is responsible for the physical and presentational aspects of the artifact.

4.3.4. Layers

The above scheme further provides an effective way to deal with complexity. This is because it superimposes three different layers upon the artifact space, each dealing with a separate type of contribution. In Figure 20, we redraw the information given in Figure 17 to clarify the contributions of the agents involved. With this additional structure, the principal contribution relations could be used to find out information about the structure of authority and power arising from agents' contributions in the RE process, as well as many other forms of selective personnel-based RT.
Part III: Approach

4.4. Qualification

With a little more information about the nature of the contributions in each of the above capacities, we can uncover further details like the mode, degree, and status of the contribution. This can provide extra material for describing more intricate contribution structures. Although numerous sets of attributes could be suggested for qualifying each of these capacities, the most useful attributes are likely to be those that can be automatically obtained as a by-product of the approach and the basic information that it collects, those we later suggest for qualifying the author capacity being a particular example. In Figure 21, we provide an example set of attributes for each capacity, then go on to illustrate what further information they can bring.

**Figure 20**: Using three contribution layers to organise the model of the contribution structure. For Artifact 1, Agent A is principal, Agent A is author, Agents C and D are documentors. For Artifact 2, Agents A and C are principals, Agent B is author, Agent B is documentor. For Artifact 3, Agents A, B, C and D are principals, Agent D is author, Agent D is documentor.
Figure 21: Example attributes to qualify the three capacities of the contribution format. Note that: (a) the attributes for qualifying the author capacity could be automatically determined based on knowledge of the artifact-based RT relations; (b) the principal capacity could either be automatically determined from signatures and circulation lists, else manually provided by the principal or others; and (c) the documentor capacity would probably need to be manually provided by the documentor.

4.4.1. Of Principal

Signatures are all-pervasive in systems and software development. They signify authorisation and the transfer of commitments between agents. They further serve to temporarily stabilise the artifact concerned and often act as measurement for project milestones. Signatures also function as claims of reliability, as declarations of adequacy, and often have the ability to make one legally responsible or accountable [Wheeler 1969]. Therefore, the principal capacity could be qualified to reflect sign-off procedures, so indicating whether the artifact is:

(1) **Approved** by the principal.

(2) **Pending approval** by the principal.

(3) **Not approved** by the principal.

A signatory-related set of attributes would be one way to qualify the principal capacity. These
three attributes could be identified automatically where formal procedures are in place amongst project documentation practices to obtain signatures, else identified by a third-party where distribution and approval lists are used. The extra information that such qualification provides could be used for identifying those points in a project at which requirements become stabilised or rejected, and under who's authority.

4.4.2. Of Author

The notion of "intertextuality" refers to how texts are related to each other [De Beaugrande & Dressler 1981] and so can be used to show how the production of a given text depends on other texts. The author capacity could be qualified to reflect the connections that the artifact in question has to other artifacts, thus revealing its authorial status. This is discussed further in Section 6.4.3, where we discuss the relations between artifacts, but is summarised for now. There are two possible scenarios:

(1) No connections exist. This means that the artifact in question is not dependent on any other artifacts, so the authorial status is that of Creator.

(2) Connections exist. This means that the artifact in question is dependent on other artifacts, so the authorial status is determined relative to the broad communicative function of each of these dependencies, and can further be distinguished by the narrower communicative purpose of the dependency.

A set of attributes based on original and derived notions of authorship would be one way to qualify the author capacity. The extra information that such qualification provides could be used for identifying trails of authorial dependency between specific agents throughout a documented requirement's life, along with the nature of these dependencies.

4.4.3. Of Documentor

Where information is derived from some other agent other than the original transmitter, the commitment with which its relayed, either verbally or in written form, generally varies. In some languages, where it is a requirement that the personal commitment with which an assertion is produced and transmitted be specified, it is important to differentiate between the source of the locution and the source of the illocution, further providing a connection of some sort between the two. In linguistic terms, it is essential to capture the mood of the transmission to determine the different degrees of commitment of the respective parties. Therefore, the documentor capacity could be qualified to reflect the mood of the documentor, say based upon those mood types given in [Matthews 1965 (cited in [Drew & Wootton 1988])], so indicating whether the documentor is:

(1) Certain that the content of the artifact is true. This reflects an emphatic mood, where the
transmitter indicates certainty and commitment to the truth of what is transmitted.

(2) **Believes** that the content of the artifact is true. This reflects a *period* mood, where the transmitter indicates belief that what is transmitted is true.

(3) **Indifferent** to the truth value of the artifact’s content. This reflects both *quotative* and *report* moods. The former means that the transmitter indicates that what is transmitted comes from a second-hand and indefinite source. The latter means that the transmitter indicates that what is transmitted comes from a second-hand and known source.

(4) **Uninformed** about the truth value of the artifact’s content. This reflects both *indefinite* and *question* moods. The former means that the transmitter indicates that they do not know whether or not what is transmitted is true and believes that the recipients also do not know. The latter means that the transmitter indicates that they do not know whether or not what is transmitted is true, but believes that the recipients do know.

A mood-related set of attributes would be one way to qualify the documentor capacity. These four attributes could be identified semi-automatically, depending on the other capacities the documentor occupies in relation to an artifact, else from rules relating to mood shifting within and between agents when information is duplicated or reused. The extra information that such qualification provides could be used to identify shifts in commitment when information about a requirement is passed on between agents or to identify points in a requirement's life where the information is more likely to become prone to inadvertent error.

### 4.5. Example of a Rudimentary Contribution Structure

The contribution format captures three aspects of a contribution. These are the principal, author, and documentor. Each of these capacities can be assumed by either an individual or a group. They can even be occupied by the same agent. In the simplest scenario, the contribution format defines the contribution structure of a primitive artifact which is not related to other artifacts. How we depict such a contribution format is repeated in Figure 22 for description purposes.
Figure 22: The contribution format of a primitive and isolated artifact defines its contribution structure.

Extending this, the sum of all the contribution formats and their interrelations defines the contribution structure of a composite artifact, again where it is not related to other artifacts. This situation is shown in Figure 23.

Figure 23: The contribution formats of all its internal components and their interrelations defines the contribution structure of a composite artifact. For clarity in this diagram, note that we do not show the interrelations between the agents themselves which arise as a consequence of the relations between the artifact and its components. This is something we return to in Section 6 when we discuss the impact of artifact-based RT relations on the definition of contribution structures.
Part III: Approach

The above description only discusses a rudimentary contribution structure. Even if the capacities of the contribution format were qualified with further attributes, we can only learn a little about the contribution structure underlying this composite artifact. The next obvious step is to consider the relations, not only within a composite artifact, but between artifacts themselves. This is the subject of Section 6.

5. Developing Contribution Structures

In this section, we describe how the information captured using the above scheme can be manipulated to infer richer details about the contribution structure. We first explain how this basic information can be used to determine details about the social roles of contributing agents and their resulting role relations. We then explain how this, in turn, imparts details about the individual and collective commitment of agents to artifacts, as well as about their social commitments to each other. These three commitment types are based on [Castelfranchi 1993] and are each described as encountered below. Finally, we give an example to demonstrate how these richer details about the contribution structure underlying requirements artifacts can be developed using these simple extensions.

5.1. Uncovering Hidden Details

When an agent contributes to an artifact in one of the above capacities, they also act in a social role. There are many suggestions as to the kinds of role that need to be assumed in systems and software to define the responsibilities and relations of personnel, and typically include roles like: (a) information systems director; (b) information systems manager; (c) systems manager; (d) programming manager; (e) operations manager; (f) database administrator; (g) systems analyst; (h) applications programmer; (i) systems programmer; (j) computer operator; (k) librarian; (l) data preparation supervisor; and (m) data preparation operator [Martin & Powell 1992]. However, these roles tend to be institutional, prescriptive, and coarse-grained. They are not really designed to reflect the changing orientations of those involved throughout the process.

In contrast, social roles are not defined a priori, but are defined relative to specific social situations, and so relative to an agent's relational ties to other agents. Social role assignment is thus accomplished interactionally and dynamically and these roles further adapt to account for the changing alignment of agents to artifacts. So, although it may be clear who the documentor of an artifact is, whether they are documenting for themselves or on the behalf of others, and how this changes with the development of the artifact concerned, is not immediately clear. Whether agents act as direct contributors, indirect contributors, mediators, third-parties, and so forth, is important information to uncover because, as mentioned earlier, agents take on different degrees of commitment when acting freely or on behalf of others. However, social relations are something which are often tricky to uncover, as hinted at in Box 13. We therefore suggest that an analysis of
social roles provides a way to get a handle on the social relations developed and sustained between agents in the RE process.

"Social bonds of some kind establish themselves at the very outset of an occasion, but may alter in scope, intensity and duration, and in the way they develop or fade. Ties of some kind cover all those present, but may coexist with others between sets, subsets, and pairs of individuals. They may be symmetrical or asymmetrical, instigated mutually or one-sidedly, and may be so independently of what obtains on other occasions or in other settings."

Box 13: Describing the multiple and changing nature of social relations (taken from [Burns 1993]).

5.2. Foundations for Further Analyses

Levinson points out that, when an agent "speaks" in the analytical capacities defined by Goffman, they are also active in a particular social role from which the words take their authority [Levinson 1988]. Levinson maintains that these social roles need to be distinguished because, whereas an agent's analytical capacity is likely to remain relatively constant, the social role in which they are active is likely to alter rather more frequently. This has associated implications for the granularity of any analysis that is made possible. Furthermore, knowledge of such roles provides essential information about person and social deixis, so can assist in the interpretation of communicated information [Levinson 1983].

It is Levinson's extensions to Goffman's work that we are mainly concerned with in this section, in particular, the distinctions he makes between basic and derived production roles. He regards Goffman's three capacities as basic production roles. He then proceeds to suggest numerous ways in which these could be re-assembled to derive more purposeful derived production roles which can reflect those attended to, and distinguished in, actual language use. His particular example that we are interested in is duplicated in Table 10 and, although we include the distinction he makes between participant and non-participant production roles for completeness, we only focus on the former in this thesis.
### Table 10: Derived production roles or social roles (taken from [Levinson 1988]).

<table>
<thead>
<tr>
<th>Participant</th>
<th>Transmitter</th>
<th>Message origin</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Source of motive</td>
<td>Source of form</td>
</tr>
<tr>
<td>Participants</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Author.</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>&quot;Ghostee&quot;.</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Spokesman.</td>
<td>✓</td>
<td>✓</td>
<td>x</td>
</tr>
<tr>
<td>Relayer.</td>
<td>✓</td>
<td>✓</td>
<td>x</td>
</tr>
<tr>
<td>Devisor.</td>
<td>✓</td>
<td>x</td>
<td>✓</td>
</tr>
<tr>
<td>Sponsor.</td>
<td>✓</td>
<td>x</td>
<td>✓</td>
</tr>
<tr>
<td>&quot;Ghostor&quot;.</td>
<td>✓</td>
<td>x</td>
<td>✓</td>
</tr>
<tr>
<td>Non-participants</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ultimate source.</td>
<td>x</td>
<td>x</td>
<td>✓</td>
</tr>
<tr>
<td>Principal.</td>
<td>x</td>
<td>x</td>
<td>✓</td>
</tr>
<tr>
<td>Formulator.</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

#### 5.3. Social Roles and Role Relations

Following Levinson, we distinguish between the basic contribution roles that are defined by the three capacities in the contribution format, and a set of derived contribution roles that can be determined from these basic ones. These derived contribution roles are the social roles that agents occupy whilst contributing to artifacts and are somewhat akin to Banton's transient roles which we mentioned in Part II. As agents move rapidly between different social roles when contributing in RE, this information can be used to develop an understanding as to how they construct and shift identities and mutual alignments over time, as well as help track finer-grained types of contribution. A simple example of such social roles, based on Levinson's example in Table 10, is shown in Figure 24.
The relationships that exist between the agents themselves when assuming these social roles with respect to artifacts, either direct relations through collaboration on identical artifacts or indirect relations through working on related artifacts, provides information about the role relations that have been dynamically formed in practice. The different types of role relation that are formed in a project tell us something about how the agents involved work together. The different agents that assume the social roles in these role relations tells us something about which agents work together and in what ways. Such information is critical for understanding the workings of the social structure, as the same two agents can interact together in many social roles, with what is done in one role relation having repercussions in many others. If we now compare Figure 25 with Figure 20, we can see that social roles provide an even more effective way to selectively delimit the contributions of agents. This is because it characterises the total nature of an agent's contribution with respect to an artifact more clearly and so is more suitable for selective forms of reasoning.
Figure 25: Using seven contribution layers to organise the model of the contribution structure. For Artifact 1, Agent A is devisor, Agents C and D are relayers. For Artifact 2, Agents A and C are sponsors, Agent B is representative. For Artifact 3, Agents A, B, C and D are sponsors, Agent C is true author.

5.3.1. Implications

Distinguishing the social roles and role relations of agents is important for a number of reasons:

(1) The notion of "social role" is central to the study of social structures [Nadel 1957]. They provide a handle with which to explore the network of relations that exist between RE participants using forms of social network analysis, like those described in [Scott 1991]. In turn, this information could be used to investigate any possible disparity between the influence of different social roles in a project with the influence of the agents filling these roles in accordance with their organisational position. It could also be used to examine an agent's social role set for a project, as well as their changing role relations to other agents. Such analyses could inform about local power and solidarity, recurrent or occasional collaboration amongst agents, emerging group alignments, substitute agents, agent reciprocation, and so forth.

(2) Social roles and role relations can also reveal information that can assist with issues relating to social accountability [Buttny 1993]. This is because the social role of an agent delineates exactly what aspects of an artifact they are committed to. This information could be used to determine the totality of who to involve or inform about changes to certain aspects of a requirement throughout its life or to identify appropriate agents when there are different...
types of query.

(3) Social roles and role relations enable and constrain the discursive options of RE participants, so can be used to reveal information to assist with issues relating to communicative competence [Hymes 1972b].

(4) Knowledge of social roles and role relations can help to deal with issues relating to agent mobility and substitution. For instance, agents who occupy the same social roles and have similar role relations exhibit a form of structural equivalence, so are more likely to be interchangeable.

5.4. Commitment

When agents assume a social role, they take on commitments. As mentioned in Section 3.3.7, we examine these commitments because the notion of commitment is a fundamental concept which underlies the examination of other notions like responsibilities, obligations, and so forth. Our interpretation of the relationship between such notions is described in Figure 26.

![Figure 26: How some overloaded terminology is being used in this thesis.](image)

In Table 11, and as a simple example, we clarify the obvious commitments that can be drawn from
the social roles of Figure 24. We then describe how the identification of social roles can be used to determine:

(1) An agent's individual or collective commitment to an artifact.

(2) An agent's social commitment to other agents.

<table>
<thead>
<tr>
<th>Role</th>
<th>Committed to: physical appearance</th>
<th>Committed to: anticipated or realised effect</th>
<th>Committed to: structural form</th>
<th>Committed to: semantic content</th>
</tr>
</thead>
<tbody>
<tr>
<td>True author</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Devisor</td>
<td>✗</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Relayer</td>
<td>✔</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Sponsor</td>
<td>✗</td>
<td>✔</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Representativ</td>
<td>✔</td>
<td>✗</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Nominal</td>
<td>✔</td>
<td>✔</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Ghost</td>
<td>✗</td>
<td>✗</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

*Table 11: An example set of rudimentary commitments to artifacts. These depend on an agent's social role when contributing to an artifact.*

5.4.1. Individual and Collective Commitment

Knowledge of an agent's social role with respect to an artifact tells us about those aspects of an artifact that specific agents can be called to account for. Individual commitment arises from a contribution relation between an agent and an artifact when the agent concerned is an individual, whereas collective commitment arises when the agent concerned is a group. Such information is directly useful for filtering the retrieval of agent sources to reflect particular types of query or change proposal. It can be used to indicate which agents to involve or inform about changes to certain aspects of a requirement. It can help directly locate the primary source of motive behind an artifact, subsequent sources of format, and so on. It therefore makes the contact point for specific aspects of an artifact explicit, so traceable as the artifact evolves.

5.4.2. Social Commitment

The social role relations that arise in practice tells us about the ensuing social commitments that are formed between agents, something that is rarely captured by formal organisational structures and pre-assigned project roles. Such relations may be established in many ways, like through direct collaboration of agents on the same artifact, or through separate contributions of agents on
artifacts which are related. The type and intensity of these role relations, and how they vary with respect to different artifacts or over time throughout a project, can provide useful material for the analysis of informal organisational structures and project roles.

5.4.3. Implications

Highlighting an agent's commitments is important for a number of reasons:

(1) It brings into play the useful analytical concept of sovereignty [Gerson 1976]. The sovereignty of any delimitable social object, here a requirements artifact, refers to the overall organisation of their associated commitments in the various settings in which they participate. This can therefore help in the management of distributed RE efforts as it tells us about self-reliance.

(2) It provides for visibility of multiple perspectives and dependencies. The former is important to help in the alignment of perspectives. The latter is important to examine how the webs of commitment associated with an agent make the concepts associated with the artifacts they have contributed to stable. Furthermore, because contribution structures could ultimately be modelled as webs of different types of commitment.

(3) The notion of commitment is computational, so can be used for analysis purposes, like investigating conventions to deal with changing commitments when requirements artifacts change or evolve. In the longer-term, these could be used to guide the development and refinement of requirements artifacts.

5.5. Example of Further Analyses

To illustrate these extensions, consider the scenario of Figure 27. Olly decides *the sensor needs to be polled once every twenty microseconds* and informs Dave. Dave writes this down, as dictated, as a requirement in the RS. Here, Olly is both the principal and the author of the written requirement, whereas Dave is its documentor. From this, we can see that Olly and Dave stand in a devisor/relayer role relation with respect to requirement G.5.7.2. As the relayer of the requirement, all Dave is committed to is its physical appearance, so he can deal with any typographical queries or change requests. Queries like: "Why twenty microseconds?", or change proposals like: "Why not make it once every ten?", need recourse to Olly. This is because she is committed to the actual content and she is the one whose position would be challenged by any change.
... and the sensor needs to be polled once every twenty microseconds

... requirement G.5.7.2 ...

Dave is relayer of requirement G.5.7.2

Olly is devisor of requirement G.5.7.2

Figure 27: A devisor/relayer role relation.

Now, if Olly and Dave stand in a devisor/relayer relationship for all the requirements in the RS that they jointly contribute to, the information that can be inferred about the role relation and commitments between these two agents will contrast with that inferred if this were only a once-off. For instance, if all the previous role relations between these two agents saw Dave as the devisor and Olly as the relayer, then this could signal a potential turning point in their working relationship, with possible knock-on effects regarding their previous and current commitments to both the project and to each other. By adopting such an approach, a variety of social details can be disclosed that would otherwise remain hidden. Such details could be used to identify developing problem areas in a project and so instigate early remedial action as appropriate. They could also be used to identify the most long-standing and effective working partnerships.

6. Relating Artifacts

So far, we have presented a scheme that can be used to define a rudimentary contribution structure underlying requirements artifacts, and have further shown how more intricate details can be obtained to develop a richer picture of this contribution structure. However, we have not yet taken into account the fact that many relations generally exist within and between the requirements artifacts themselves, namely those links that provide for conventional forms of artifact-based RT. In this section, we explain why it is essential to account for these artifact-based RT relations, and
explain their impact on defining contribution structures. We propose a categorisation for the different types of artifact-based RT relation, describe each category, and indicate how knowledge of these various relations can be used to obtain a more detailed picture of the contribution structure. In so doing, we focus mainly on those specific kinds of artifact-based RT relation currently not considered for RT purposes, as well as indicate why they are likely to have the most impact on the contribution structure. Finally, we provide an example to clarify how the artifact-based RT relations we describe can combine to uncover hidden details about the contribution structure underlying RE.

6.1. Accounting for Artifact-Based RT Relations

When modelling those agents who have contributed to the artifacts produced in RE, it is important to take into account how these artifacts themselves are related to each other. This is because both the production and reception of a given artifact usually depends, either directly or indirectly, on an agent’s awareness of other artifacts. So, if one requirements artifact is a subsequent specialisation of another, it seems reasonable to assume that some responsibility for the resulting artifact will often be retained by the original contributor(s). In a purely textual domain, and one we shall return to later on in this section, this observation has been referred to as "intertextuality" [Culler 1981, De Beaugrande & Dressler 1981], and is explained in Box 14. Our approach therefore needs to recognise, and deal with, the relations that exist within and between the artifacts themselves when defining contribution structures, particularly if it is to distinguish original artifacts from artifacts that have been duplicated or derived from others.

"... literary works are to be considered not as autonomous entities, 'organic wholes', but as intertextual constructs: sequences which have meaning in relation to other texts which they take up, cite, parody, refute, or generally transform. A text can be read only in relation to other texts, and it is made possible by the codes which animate the discursive space of a culture."

Box 14: An explanation of the term "intertextuality" (taken from [Culler 1981]).

6.2. Categories of Artifact-Based RT Relation

We suggest that there are three broad categories of artifact-based relation which are useful for RT. These describe alternative artifact-based structures and provide the basis for different types of RT. They are:

(1) Temporal relations, used to reflect the chronological order in which requirements artifacts are produced.
(2) **Developmental relations**, used to reflect the logical order in which requirements artifacts are produced.

(3) **Auxiliary relations**, used to reflect many additional and supplementary forms of order between requirements artifacts.

The first two categories of relation depict the domineering lines of project structure, so capture the *macrostructure* of the RE process. The relations typically used in these categories are fairly well established and are used to provide more conventional forms of artifact-based RT. The third category of relation depicts the subordinate and supporting lines of project structure, so captures the *microstructure* of the RE process.

In this section, we are mainly concerned with developing a suitable set of relations within this latter category, then describing what they bring to the approach. This is because: (a) there is no well established set of such relations in use; (b) we suggest that it is these relations that have subtle, though crucial, effects on the definition of contribution structures; and (c) we could propose a set of auxiliary relations which subsumes the others and so encompasses those relations used for more conventional forms of RT. For completeness, we briefly describe the first two categories, and point to the sort of details that they enable us to infer about those involved in RE. It is worth pointing out that, to date, such temporal and developmental relations have not really been used for such purposes. We thereby demonstrate how it is possible to exploit existing practice in RT to reveal potentially useful details about the contribution structure underlying RE.

### 6.2.1. Temporal Relations

Temporal relations are based on some notion of time stamps. These relations are often formed as a result of making records about when work on an artifact was formally instigated and formally signed-off, when an artifact was registered in a project repository and subsequently altered, and so forth. The resulting structure is generally a coarse-grained linear sequence of artifacts, depending on the granularity of the artifacts for which such relations are defined, though any concurrency and parallelism that occurs can generally be represented. They therefore model the chronological order in which requirements artifacts are produced and so provide the means to trace requirements history. In a textual domain, such relations would correspond to the *syntagmatic* relations which describe how the basic units of a semantic analysis occur in sequence.

To enable us to reason with the resultant historical structure, we need to provide these relations with some semantics. For instance, and as illustrated in Figure 28, they could be augmented with predicates from a logic based upon temporal intervals [Allen 1983, Allen 1984]. Here, these could be used to reflect the order and duration of a requirement artifact's development, and further predicates and axioms could also be specified to define additional behaviour.
Figure 28: Historical structure and historical RT. For clarity in this diagram, we only model the relationship between whole requirements artifacts, so it would yield a trace of a requirement's history at a relatively coarse level. For finer-grained historical details and RT, the components of the individual requirements artifacts could themselves be modelled and described in the same manner.

Such relations can inform about the temporal relations between contributing agents, as well as between the requirements artifacts themselves. This allows us to quantify the extent of an agent's involvement in a project over time. They can tell us whether certain agents have been involved continuously throughout the whole project, with how many artifacts, and whether this involvement has been in the same or changing capacities. When these agents have not been involved, they can tell us who has taken over. When working arrangements have changed, these relations can be used to retrieve the artifacts produced within corresponding intervals, so are useful for reducing the search space when retrieving those artifacts an agent may have contributed under different conditions. The temporal relations between a specific agent's set of contributions can also be used to examine how their understanding of a requirement has evolved over time. In this way, temporal relations can be used to tell us about those agents who make up the backbone of a project, those who make up the periphery, those who dip in now and again to offer expertise or back-up, those who are the earliest surviving members of a project team, and those who are the most recent members. They are not only a pointer to the longevity and breadth of an agent's contributions.
within a project, but a pointer to the kinds of project information that different agents are likely to hold.

Furthermore, as the duration of a project is a critical path, temporal relations can also be used to tell us about who is, or was, a contributor to those artifacts in the critical path and how. Used during the RE process, we can identify concurrent artifacts which rely, say, on the same agent to document each. In this way we can highlight where time problems are likely to occur, prioritise work if one of them lies on the critical path, suggest alternative documentors, else leave them to concur where not critical. We can further use temporal relations to identify areas where it may be possible to speed up the project by suggesting changes to working arrangements, or to suggest ways to improve the quality of the project by ensuring the most suitable agents participate in those artifacts which constitute the critical path. Used from a retrospective point of view, we can examine factors related to best practice, like the most effective working arrangements with regard to factors such as the timely delivery of project milestones.

6.2.2. Developmental Relations

Developmental relations are used to model the development strategy, or what is often referred to as requirements flow-down, thereby illustrating how one requirement artifact realises another in order to progress a project. As with the temporal relations, the relations represented here can be between whole artifacts, yielding coarse-grained RT, or between any components, yielding finer-grained RT. In a textual domain, such relations would correspond to the paradigmatic relations which describe how the basic units of a semantic analysis can substitute for one another.

Developmental relations are characteristically defined in terms of composition and decomposition, the precise semantics being dictated by individual development methods or approaches to RT. These often include the use of parent-child relationships [Sodhi 1992], predecessor-successor relationships [Watkins & Neal 1994], amongst other variants. The resultant development structure is generally hierarchical in nature and represents any composition and decomposition that has taken place to artifacts in the course of a project. Any intermediate or subordinate artifacts are often appended to the corresponding dominating artifacts in some manner, though not often taken into account for RT purposes. Figure 29 illustrates how these developmental relations describe the logical structure of development and provide the means to trace requirements flow-down.
Such relations can inform about the developmental relations between contributing agents, as well as between the requirements artifacts themselves. This allows us to quantify the longevity and breadth of an agent's contributions with respect to a particular requirement throughout its exploration and subsequent development. They can tell us whether certain agents have contributed continuously to a requirement as it has progressed from its original source to its final form, whether their contribution has been in a consistent or changing capacity, and who else has periodically or consistently contributed. As these relations are also commonly used to signal the versioning of artifacts, they can be used to reveal the contribution structures relating to the different versions and configurations of artifacts. In this way, developmental relations can be used to provide pointers as to which agents are likely to be the sources of specific information concerning a requirement at different stages in its development, and can offer the potential to monitor a requirement's progression to ensure that some common thread of involvement is maintained.

Developmental relations can also be used to give insight as to the probable communication requirements between agents. For instance, there is likely to be more need to communicate with those agents who were involved with a specific requirement in a previous version or at a more abstract level of its life, than with those agents whose contributions are not developmentally
related. There is also likely to be a need to communicate with those agents working in the same developmental stage, say with logically related requirements, in situations where proposed changes need to be assessed. In addition, these relations can provide for crude error checking facilities if used to track for logical inconsistencies in the development process. For instance, what signals should be flagged when an agent who has acted in a relayer's role throughout the exploration of a requirement suddenly becomes its devisor, or when the roles of two agents become reversed in such a progression?

When the temporal relations are combined with the developmental relations, more can be analysed about contribution structures. For example, we can examine how the contributors, social roles, role relations, and commitments change in time with respect to a specific requirement's development path. Such information can be used to explore how significant the changing relations of an agent to a requirement is, how significant the amount of change-over is, and so on.

6.2.3. Auxiliary Relations

Auxiliary relations describe the additional ways in which information both within and across requirements artifacts is related. It is exactly these type of relation that can deal with the traceability of keywords and cross references amongst requirements artifacts. They thereby provide supplementary structures and forms of RT. It is the intricacy of these network-like structures that determines the scope and granularity of the additional RT that is made possible. Here, we are concerned with those auxiliary relations that have implications for the definition of contribution structures. The two types of auxiliary relation we are most concerned with are what we call containment relations and connectivity relations. Examples of these, along with their influence on the definition of contribution structures, are given in the following two sub-sections and are illustrated in Figure 30.
6.3. Containment Relations

Containment relations delimit the internal structure of composite artifacts, as illustrated in Figure 31. If we ignore external artifacts connected to these composite artifacts for now, the contribution structure of a composite artifact is described by all its component contribution formats and all their interconnections. By recording the relation between a composite artifact and those other artifacts which are its components, we can thereby make the task of assigning the contribution format much easier. Though clearly a composite artifact may have different agents acting in identical capacities with respect to its components, it can be a default assumption that they are the same, until explicitly declared otherwise. Therefore, the use of containment relations means that multiple contribution formats can be defined, interrelated, and managed for an artifact. Areas of contribution can become more finely delineated over time as changes are made by various agents and as artifacts are versioned, partitioned, and used elsewhere. The resultant layering effect can be instrumental in managing the complexity of large numbers of contributors. As a containment relation is purely structural in nature, we see no further leverage to be gained by giving this
relation finer-grained semantics, like clarifying the precise reason for the containment.

Figure 31: The layered contribution formats of a composite artifact. Note how the containment relations serve to layer the contribution formats, suggesting default incumbents of the various capacities in so doing.

6.3.1. Example of Use

In Figure 31, the agents who are the incumbents of the capacities P(A), A(A) and D(A) are, by default, the incumbents of the capacities P(A2), A(A2) and D(A2), unless declared otherwise. Continuing in this manner, the agents who are the incumbents of P(A2), A(A2) and D(A2) are, by default, the incumbents of P(A2.1), A(A2.1) and D(A2.1). Where internal components of a composite artifact overlap, as is the case with artifact Ax, the default incumbents of the capacities are the sum of the separate overlapping artifacts, again unless declared otherwise. Note that the component artifacts may have been formed because they were declared the end-point of an artifact-based RT relation to an external artifact not shown in the figure. By delineating the contributors to such newly specified components, it is possible to keep finer track of the contributors to these specific components as they are used and developed elsewhere.
6.4. Connectivity Relations

Connectivity relations are the additional types of relations that can exist to connect artifacts and are more complex in nature than the above type of containment relation. We explain how the set of connectivity relations we propose for RT purposes is derived from a linguistics basis and is partitioned into those relations which function to reference or to adopt. We then indicate how this basis offers the ability to define these relations at progressively finer levels of granularity which, in turn, impacts the intricacy of any contribution structures that can be defined through their use.

6.4.1. Linguistic Basis

In order to inform a useful set of connectivity relations, one which highlights the different ways in which the various connections between artifacts impinges on the definition of contribution structures, we look to that work in text linguistics which examines the various ways in which textual occurrences can be related [De Beaugrande & Dressler 1981]. De Beaugrande and Dressler list seven basic ways in which texts can be related and show how these are the constitutive principles which define textual communication. They are illustrated in Figure 32 and are: (a) cohesion; (b) coherence; (c) intentionality; (d) acceptability; (e) informativity; (f) situationality; and (g) intertextuality. We focus on the relations of cohesion and coherence here, since these are purely text-centred notions, so can be used to reflect the relations between artifacts.

Cohesion relations are those which deal with how the components of a surface text are mutually dependent and stick together, so deal with connectivity at the surface. As cohesion is a feature of text, these relations are explicitly cued by grammatical dependencies, such as pronouns. Many sets of explicitly cued cohesion relations have been proposed, as seen in [Crystal 1987, De Beaugrande & Dressler 1981, Halliday & Hasan 1976, Hoey 1983].

In contrast, coherence relations are those which deal with how the components of a text are mutually accessible and relevant, so deal with connectivity of the underlying content. They deal with conceptual dependencies and are functional in nature. Work on text coherence includes: (a) theories of discourse relations, which elucidate the implicit relations that exist between sentences of a text and bind it together, as seen in the work of [Grimes 1975, Hobbs 1979]; (b) theories of discourse structure, which explain the underlying hierarchical structure of texts in terms of their basic units and ways in which they can relate, as seen in the work of [Grosz & Sidner 1986]; and (c) theories which combine the previous ideas, as seen in the work on Rhetorical Structure Theory [Mann 1984, Mann & Thompson 1983, Mann & Thompson 1988]. These distinctions are described more fully in [Knott & Dale 1993].
6.4.2. Working Set of Relations

Our set of connectivity relations draws upon the above work, which deals with connectivity at the sentential level of text, and extends the underlying concepts to encompass our definition of "artifact". They thereby indicate how the surface structures and underlying concepts are shared and borrowed within and between artifacts to promote coherence and cohesion. This can tell us about how requirements artifacts have been used amongst agents. We do not claim to have an exhaustive and conclusive set, a contentious issue anyway according to [Knott & Dale 1993], but rather a working set to examine the impact of such types of relation on defining contribution structures.

Our working set of connectivity relations is divided into two groups. These reflect the broad communicative function that a connectivity relation can serve between two artifacts. By the term "communicative function", we mean how a relation operates, in a communicative sense, with respect to each of the artifacts it relates. Based on the subsumed cohesion and coherence relations, we suggest that a connectivity relation can either function to reference or to adopt the content of another artifact. These two groups are described below and are illustrated in Figures 33 and 34. With their respective descriptions, Tables 12 and 13 show how each broad group can be more finely decomposed, to firstly account for the communicative purpose of the relation, and to
secondly show how the underlying coherence and cohesion relations have been used in this classification. These tables are interesting because knowledge of these connectivity relations, and at various levels of granularity, can impact the definition of contribution structures in subtle ways. This is because they can indicate more finely the nature of the relationships between those agents contributing to the artifacts connected by the relation.

6.4.2.1. References Connectivity Relation

Connectivity relations which function to reference exist when the physical content of the source and target artifacts does not overlap. This means that the information in the target artifact is not integral to the information in the source artifact. Instead, it has information which is either subordinate, superordinate, or coordinate to the source artifact. The referenced artifact may be explicitly signalled in the source, perhaps by cross references, keywords, or synonyms, else implicitly signalled.

![Figure 33: The "references" connectivity relation.](image)

In Table 12, we show how a referencing communicative function can be identified from the various communicative purposes the relation can serve. For instance, a references communicative function exists when the purpose of the connectivity relation is either: (a) to frame; (b) to match; (c) to substantiate; or (d) to show causality. Similarly, a relation that serves to substantiate can be identified when the information in the target artifact illustrates or supports that in the source artifact.

<table>
<thead>
<tr>
<th>Purpose of relation (reason for referencing)</th>
<th>Function of referenced artifact B with respect to referencer artifact A (showing examples of subsumed cohesion and coherence relations in brackets)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) To frame - B provides a framework in which to understand or interpret the information in A.</td>
<td>(a) B gives preparatory information for A (background, circumstance, setting, location).</td>
</tr>
<tr>
<td></td>
<td>(b) B gives motivation for A (purpose, reason).</td>
</tr>
<tr>
<td></td>
<td>(c) B is the particular case of A (component, representative, decompose, subordinate).</td>
</tr>
<tr>
<td></td>
<td>(d) B is the general case of A (abstract, compose, collect, superordinate).</td>
</tr>
</tbody>
</table>
(e) B enables the analysis made in A (argue, evaluate, explain, interpret, critique, resolve).

(f) B enables the inference made in A (induct, deduct, abduct, refute).

(2) To match - A and B are juxtaposed for a specific reason.

(a) B is compared with A (liken, relate, analogy, similarity, resemblance).

(b) B is contrasted with A (antithesis, difference).

(c) B is coordinated with A (synchronise, associate, alternative, option, branch).

(3) To substantiate - information in B strengthens that in A.

(a) B illustrates A (exemplify, demonstrate, show).

(b) B supports A (solidify, assist, consolidate, justify, evidence, backing).

(4) To show causality - cause/consequence pairs.

(a) B (causes, enables) A.

(b) A (replies to, result of, answers, responds to) B.

Table 12: Decomposition of the "references" group of connectivity relations.

6.4.2.2. Adopts Connectivity Relation

Connectivity relations which function to adopt exist when the physical content of the source and target artifact overlaps in some way. This means that the information in the target artifact is integrated into the information in the source artifact, either exactly, inexactly, in full, or in part. Furthermore, the information in the source artifact could either be a static or dynamic version of that in the target artifact, distinctions which respectively reflect the concepts of embedding and linking that are inherent in Microsoft's Object Linking and Embedding mechanism [Microsoft 1993]. If static, the content of the source would not change as the content in the target changes. If dynamic, the content of the source would change as the content in the target changes. Such distinctions are important to account for as they would signal where responsibility for future maintenance of an artifact lies and also provide the potential for requirements artifacts to be "living".

Figure 34: The "adopts" connectivity relation.
In Table 13, we show how an adopting communicative function can be identified from the various communicative purposes the relation can serve. For instance, an adopts communicative function exists when the purpose of the connectivity relation is either: (a) to copy; (b) to add to; (c) to remove from; or (d) to alter information in the target. Similarly, a relation that serves to add can be identified when the information in the source artifact defines or elaborates that in the target artifact.

<table>
<thead>
<tr>
<th>Purpose of relation (reason for adopting)</th>
<th>Function of adopter artifact A with respect to adopted artifact B (showing examples of subsumed cohesion and coherence relations in brackets)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) To copy - use existing information as is with no changes or additions.</td>
<td>A copies information that exists in B (repeat, reassure, orient, resume, emphasise).</td>
</tr>
<tr>
<td>(2) To add - use existing information with extensions to content.</td>
<td>A adds to the information that exists in B (define, describe, qualify, elaborate, develop, extend).</td>
</tr>
<tr>
<td>(3) To remove - use existing information with reduction of content.</td>
<td>A removes from the information that exists in B (delete, dismiss, reject, repudiate, subtract, replace).</td>
</tr>
<tr>
<td>(4) To alter - use existing information with changes to content.</td>
<td>A alters the wording or structure of information that exists in B (refine, summarise, rephrase, rename, clarify, correct).</td>
</tr>
</tbody>
</table>

*Table 13: Decomposition of the "adopts" group of connectivity relations.*

6.4.3. Purposes and Qualifiers

Although the broad connectivity relation types are to reference and to adopt, we have shown how the precise nature of the relation can be progressively determined through its communicative purpose, and have mentioned a number of qualifiers that are useful for clarifying the physical nature of the relation. These are summarised in Figure 35. We later explain how such details can be used to identify the connectivity relations between artifacts more precisely and to reveal subtle aspects about the relations between those contributors to the related artifacts concerned.
159

6.4.4. Example of Use

In this example, we illustrate one way in which the above connectivity relations can be used to inform the definition of contribution structures. As discussed in Section 4.4.2, the way in which we suggest we qualify the authorial status of an artifact depends on the connectivity relations that the artifact concerned has to other artifacts. There, we identified two possible scenarios:

**Figure 35: Identifying connectivity relations.**
(1) No connectivity relations exist. If the artifact being authored is not related to other artifacts with connectivity relations, then the authorial status for the artifact is that of creator. The communicative function of the artifact could be to create, introduce, initiate, propose, begin, provide premises, provide setting, identify, define, and so forth.

(2) Connectivity relations exist. If the artifact being authored is related to other artifacts with connectivity relations, then the authorial status is either that of referencer, adopter, or both. The broad communicative function of this artifact can then be determined relative to each connectivity link and corresponding artifact in order to determine this status.

In Figure 36 and in the subsequent discussion, we show how such qualification provides information about the agent chains of dependency that emanate from the artifacts they have in common, in turn enabling us to appreciate the different connotations that underlie what it means to be an author. It is worth pointing out that this qualification could be determined automatically if the taxonomy of artifact-based RT relations we described above were supported by the RT scheme in use.

In this figure, author A(2) is an adopter of author A(1)’s contribution, and author A(3) is a referencer of author A(1)’s contribution. Author A(1), however, is a creator. At a finer level of granularity, author A(2) could be qualified as an adopter(to add) if artifact 2 were a description of a concept introduced in artifact 1, whilst author A(3) could be qualified as referencer(to frame) if artifact 1 provided the background necessary for understanding the information in artifact 3. In this way, knowledge of the connectivity relations between artifacts can provide insight about both the structure and nature of agent chains of dependency in a project.
6.5. Example of Auxiliary Relations in Use

Following on from the previous example, we indicate how the containment and connectivity relations can together be used to qualify the authorial status of a contribution. In Figure 37, Olly's authorial status changes throughout artifact 1 as it is directly related to others, artifacts 2 and 3. As these relations are formed, artifact 1 becomes a composite artifact with internally delineated components 1.1 and 1.2 to reflect the source of these relations.

![Figure 37: Containment and connectivity relations in use. Together, they can be used to identify default authors, to ascertain the mode of authorship, and to clarify authorial dependencies throughout a project.](image)

Through the containment relations, we can see that Olly is the default author of the components of artifact 1. We can also see that Olly's changing authorial status across these components can be determined relative to each of their connectivity relations. With the relation to artifact 3, Olly is adopting Dave's authored contribution in artifact 3. With the relation to artifact 2, Olly is referencing Paddy's authored contribution in artifact 2. Note that the more specific the specification of communicative purpose, say Olly is adopting Dave's contribution because she is altering it, or more specifically because she is correcting it, the more discerning the authorial status can become. As a consequence, these show that, even if it is the same agent who is the author of
artifact 1 and all of its internal components, the authorial status of all of these contributions can be very different dependent on the type of connectivity relations. Such information can be used to clarify not only Olly's commitments to artifact 1, but also Olly's commitments to other agents who are indirectly related. We can thereby immediately recognise Paddy's underlying influence on artifact 1. As different levels and kinds of authorial dependencies can be traced between agents, this obviously has implications for retrieving those agents to inform or involve about, say, different types of changes proposed to the content in the different regions of artifact 1.

7. Modelling Contribution Structures

We have outlined an approach to model and subsequently make use of the contribution structure underlying requirements artifacts to address the crux of the RT problem. In this section, we present a semi-formal model of the concepts that underpin our approach to clarify the details as to how contribution structures can be defined and maintained, and also to reduce any ambiguity. In Section 8, we then use this model to clarify how contribution structures can be consistently used to enable personnel-based RT and so to answer the aforementioned frequently asked questions.

7.1. Model Preliminaries

Here, we introduce the key concepts of the model. These are the sets and data types with which the dynamic state of the contribution structure can be modelled. Note that we make use of symbols commonly used in set theory.

7.1.1. Basic Types

The basic elements with which we define contribution structures are the agents and artifacts. We therefore represent these in our model by the finite set of all possible agents \([AGENT]\) and the finite set of all possible artifacts \([ARTIFACT]\). Although we assume such sets exist, we assume nothing about their contained elements.

\[
\text{[AGENT]} \quad \text{[ARTIFACT]}
\]

7.1.2. Type Definitions

The data type definitions are used to model the basic concepts necessary to describe the fundamentals of the approach, so to model the allowable contribution capacities, artifact-based RT relations, qualifications, and so forth. This means that, for example, the type \(\text{CAPACITY}\) can more precisely be defined as \(\text{Principal}, \text{Author}, \text{or Documentor}\).
7.1.3. Derived Types

The derived types are those which make use of the above basic types and type definitions to model the agents and artifacts in specific RE exercises to determine their underlying contribution structure. We represent these in our model by the finite set of registered agents \([\text{REG}\_\text{AG}]\), the finite set of registered artifacts \([\text{REG}\_\text{ART}]\), the finite set of group agents \([\text{GRP}\_\text{AGS}]\), the finite set of contribution relations \([\text{CONTRIB}\_\text{REL}]\), and the finite set of artifact-based relations \([\text{ART}\_\text{REL}]\):

1. \([\text{REG}\_\text{AG}]\) is the finite set of all the agents registered to a project and is a subset of the set of all possible agents. We include this to delineate those agents which are direct participants in a specific project from a potentially large pool of agents in an organisational model.

\[
\text{REG}\_\text{AG: set ofAGENT} == \{ag_1,...,ag_i,...,ag_n\}
\]

2. \([\text{REG}\_\text{ART}]\) is the finite set of all the artifacts registered to a project and is a subset of the set of all possible artifacts. We include this to delineate those requirements artifacts which are held in a project repository, and so are under RT control, from a potentially large pool of project artifacts which may not be in on-line form.

\[
\text{REG}\_\text{ART: set ofARTIFACT} == \{art_1,...,art_i,...,art_n\}
\]

3. \([\text{GRP}\_\text{AGS}]\) is the finite set of all the agent groups and their members registered to a project. Each group's details is expressed using the pair \(\text{grp_details} = (\text{grp}\_\text{name,}(\text{grp}\_\text{members}))\) where \(\text{grp}\_\text{name} \in \text{AGENT}\) and \(\text{grp}\_\text{members} \in \text{set of AGENT}\). The notation \(\text{grp}\_\text{details.grp}\_\text{name}\) and \(\text{grp}\_\text{details.grp}\_\text{members}\) is used to refer to the components of
this pair. Similarly, grp_details.grp_member₁,...,grp_details.grp_memberᵢ,...,grp_details.grp_memberₙ is used to refer to each of the members of the group.

\[ \text{GRP\_AGS set of (AGENT x set of AGENT) = = \{grp\_details₁,...,grp\_detailsᵢ,...,grp\_detailsₙ\} } \]

(4) CONTRIB_REL is the finite set of all the contribution relations between agent/artifact pairs registered to a project. Each contribution relation’s details is expressed using the 4-tuple \( \text{contrib_rel} = (\text{ag,art,capacity,qualif}) \) where \( \text{ag} \in \text{AGENT}, \text{art} \in \text{ARTIFACT}, \text{capacity} \in \text{CAPACITY}, \) and \( \text{qualif} \in \text{QUALIF} \). The notation \( \text{contrib_rel.ag, contrib_rel.art, contrib_rel.capacity and contrib_rel.qualif} \) is used to refer to the components of this 4-tuple.

\[ \text{CONTRIB\_REL: set of (AGENT x ARTIFACT x CAPACITY xQUALIF) = = \{contrib\_rel₁,...,contrib\_relᵢ,...,contrib\_relₙ\} } \]

(5) ART_REL is the finite set of all the auxiliary artifact-based RT relations between pairs of artifacts registered to a project. Each artifact-based relation’s details is expressed using the 4-tuple \( \text{art_rel} = (\text{artsource,arttarget,arelation, purpose}) \) where \( \text{artsource} \in \text{ARTIFACT}, \text{arttarget} \in \text{ARTIFACT}, \text{arelation} \in \text{ARELATION}, \) and \( \text{purpose} \in \text{PURPOSE} \). The notation \( \text{art_rel.artsource, art_rel.arttarget, art_rel.arelation and art_rel.purpose} \) is used to refer to the components of this 4-tuple.

\[ \text{ART\_REL: set of (ARTIFACT x ARTIFACT x ARELATION x PURPOSE) = = \{art\_rel₁,...,art\_relᵢ,...,art\_relₙ\} } \]

7.1.4. Specification of State

In our model, we represent the state of the contribution structure for a specific project, CS, by the finite set of all its registered agents, artifacts, and group agents, along with the defined contribution relations and artifact-based RT relations. The state of the model provides all the information needed for the operation of the approach.

\[ \text{CS set of AGENT x set of ARTIFACT x set of (AGENT x set of AGENT) x set of (AGENT x ARTIFACT x CAPACITY x QUALIF) x set of (ARTIFACT x ARTIFACT x ARELATION x PURPOSE) = = \{REG\_AG, REG\_ART, GRP\_AGS, CONTRIB\_REL, ART\_REL\} } \]
7.1.5. Specification of Initial State

At the onset of a project, the derived sets are initially empty, so the state of the CS is also empty. We represent this initial state accordingly in our model.

\[
\text{CS} = \emptyset \quad \text{as} \\
\text{REG\_AG} = \emptyset \\
\text{REG\_ART} = \emptyset \\
\text{GRP\_AGS} = \emptyset \\
\text{CONTRIB\_REL} = \emptyset \\
\text{ART\_REL} = \emptyset
\]

7.1.6. Invariants

Some examples of the predicates that hold between the components of our model, for the operation of the approach, are outlined below:

1. No two registered agents can have the same identifier.

\[\forall \text{ag1,ag2} \in \text{REG\_AG} (\text{ag1} \neq \text{ag2})\]

2. No two registered artifacts can have the same identifier.

\[\forall \text{art1,art2} \in \text{REG\_ART} (\text{art1} \neq \text{art2})\]

3. Each artifact in an artifact-based relation must be a registered artifact for the relation to have valid end points.

\[\forall \text{art\_rel} \in \text{ART\_REL} (\text{art\_rel.art}_{\text{source}} \in \text{REG\_ART} \land \text{art\_rel.art}_{\text{target}} \in \text{REG\_ART})\]

4. Each artifact in a contribution relation must be a registered artifact for the relation to have a valid contribution.

\[\forall \text{contrib\_rel} \in \text{CONTRIB\_REL} (\text{contrib\_rel.art} \in \text{REG\_ART})\]

5. Each agent in a contribution relation must be a registered agent for the relation to have a valid contributor.
∀ contrib_rel ∈ CONTRIB_REL (contrib_rel.ag ∈ REG_AG)

(6) A group agent cannot have itself as one of its members.

∀ grp_details ∈ GRP_AGS (grp_details.grp_name \notin grp_details.grp_members)

(7) A registered artifact cannot be related to itself to prevent circularity.

∀ art_rel ∈ ART_REL (art_rel.art.source ≠ art_rel.art.target)

(8) If a contribution relation is qualified, this must be according to the specified attributes for qualification.

∀ contrib_rel ∈ CONTRIB_REL
  if contrib_rel.capacity = Principal then contrib_rel.qualif ∈ PQUALIF
  if contrib_rel.capacity = Author then contrib_rel.qualif ∈ AQUALIF
  if contrib_rel.capacity = Documentor then contrib_rel.qualif ∈ DQUALIF

(9) If an artifact-based relation is more finely described, this must be according to the specified categories for communicative purpose.

∀ art_rel ∈ ART_REL
  if art_rel.relation = Contains then art_rel.purpose ∈ CPURPOSE
  if art_rel.relation = References then art_rel.purpose ∈ RPURPOSE
  if art_rel.relation = Adopts then art_rel.purpose ∈ APURPOSE

7.2. Specification of Operations to Alter State

The initial state of the CS at the onset of a project has already been described. Here, we describe those operations that define how the state of the CS can be dynamically instantiated and maintained to model our approach. These specify how the contributions, contributors, and their interrelations, can be added and removed. For each operation that changes the state of the CS, we describe its name, signature, pre-conditions, and post-conditions. In Section 8, we go on to describe those operations which act on the knowledge of the current state of the CS to make use of the resultant information for personnel-based RT purposes.
7.2.1. Adding Artifacts, Agents and Relations

Those operations which serve to add information about artifacts, agents, relations, and so forth, to define and maintain an up-to-date model of the contribution structure, are given below:

(1) Operation to add agents to the set of registered agents.

<table>
<thead>
<tr>
<th>Name: add_agent(ag)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sig:</strong> AGENT → (REG_AG → REG_AG)</td>
</tr>
<tr>
<td><strong>Pre:</strong> ag ∉ REG_AG</td>
</tr>
<tr>
<td><strong>Post:</strong> REG_AG' = REG_AG ∪ {ag}</td>
</tr>
</tbody>
</table>

(2) Operation to add artifacts to the set of registered artifacts.

<table>
<thead>
<tr>
<th>Name: add_artifact(art)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sig:</strong> ARTIFACT → (REG_ART → REG_ART)</td>
</tr>
<tr>
<td><strong>Pre:</strong> art ∉ REG_ART</td>
</tr>
<tr>
<td><strong>Post:</strong> REG_ART' = REG_ART ∪ {art}</td>
</tr>
</tbody>
</table>

(3) Operation to add groups to the set of registered group agents.

<table>
<thead>
<tr>
<th>Name: add_group(grp_name, grp_members)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sig:</strong> (AGENT × set of AGENT) → (GRP_AGS → GRP_AGS)</td>
</tr>
<tr>
<td><strong>Pre:</strong> grp_name ∈ REG_AG</td>
</tr>
<tr>
<td>∀ ag ∈ grp_members (ag ∈ REG_AG)</td>
</tr>
<tr>
<td>(grp_name, grp_members) ∉ GRP_AGS</td>
</tr>
<tr>
<td><strong>Post:</strong> grp_details_{n+1} = (grp_name, grp_members)</td>
</tr>
<tr>
<td>GRP_AGS' = GRP_AGS ∪ {grp_details_{n+1}}</td>
</tr>
</tbody>
</table>

(4) Operation to add group members to existing group agents.

<table>
<thead>
<tr>
<th>Name: expand_group(old_grp_name, new_grp_members)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sig:</strong> (AGENT × set of AGENT) → (GRP_AGS → GRP_AGS)</td>
</tr>
<tr>
<td><strong>Pre:</strong> old_grp_name ∈ REG_AG</td>
</tr>
<tr>
<td>∀ ag ∈ new_grp_members (ag ∈ REG_AG)</td>
</tr>
<tr>
<td>∃ grp_details_i ∈ GRP_AGS s.t. (grp_details_i, grp_name = old_grp_name) ∧</td>
</tr>
<tr>
<td>new_grp_members ∉ grp_details_i.grp_members</td>
</tr>
<tr>
<td>existing_members = grp_details_i.grp_members</td>
</tr>
<tr>
<td><strong>Post:</strong> GRP_AGS = GRP_AGS replacing grp_details_i with</td>
</tr>
<tr>
<td>(old_grp_name, existing_members ∪ new_grp_members)</td>
</tr>
</tbody>
</table>
(5) Operation to add contribution relations between agents and artifacts.

**Name:** add_contrib_rel(\(ag,art,capacity,qualif\))

**Sig:** \((AGENT \times ARTIFACT \times CAPACITY \times QUALIF) \rightarrow (CONTRIB\_REL \rightarrow CONTRIB\_REL)\)

**Pre:**  
\(ag \in REG\_AG\)  
\(art \in REG\_ART\)  
\(capacity \in CAPACITY\)  
\(if\ capacity = \text{Principal} then qualif \in PQUALIF\)  
\(if\ capacity = \text{Author} then qualif \in AQUALIF\)  
\(if\ capacity = \text{Documentor} then qualif \in DQUALIF\)  
\((ag,art,capacity,qualif) \notin CONTRIB\_REL\)

**Post:**  
\(\text{contrib}\_rel_{n+1} = (ag,art,capacity,qualif)\)  
\(CONTRIB\_REL' = CONTRIB\_REL \cup \{\text{contrib} \_rel_{n+1}\}\)

(6) Operation to add artifact-based relations between artifacts.

**Name:** add_art_rel(\(art\_source,art\_target,arelation,purpose\))

**Sig:** \((ARTIFACT \times ARTIFACT \times ARELATION \times PURPOSE) \rightarrow (ART\_REL \rightarrow ART\_REL)\)

**Pre:**  
\(art\_source \in REG\_ART\)  
\(art\_target \in REG\_ART\)  
\(art\_source \neq art\_target\)  
\(arelation \in ARELATION\)  
\(if\ arelation = \text{Contains} then purpose \in CPURPOSE\)  
\(if\ arelation = \text{References} then purpose \in RPURPOSE\)  
\(if\ arelation = \text{Adopts} then purpose \in APURPOSE\)  
\((art\_source,art\_target,arelation,purpose) \notin ART\_REL\)

**Post:**  
\(\text{art} \_rel_{n+1} = (art\_source,art\_target,arelation,purpose)\)  
\(ART\_REL' = ART\_REL \cup \{\text{art} \_rel_{n+1}\}\)

7.2.2. Deleting Artifacts, Agents and Relations

Those operations which serve to remove information about artifacts, agents, relations, and so forth, to define and maintain an up-to-date model of the contribution structure are given below:
(1) Operation to delete agents from the set of registered agents.

**Name**: del_agent\( (ag) \)

**Sig**: \( \text{AGENT} \rightarrow (\text{REG\_AG} \rightarrow \text{REG\_AG}) \)

**Pre**: \( ag \in \text{REG\_AG} \)

**Post**: \( \text{REG\_AG}' = \text{REG\_AG} \setminus \{ag\} \)

(2) Operation to delete artifacts from the set of registered artifacts.

**Name**: del_artifact\( (art) \)

**Sig**: \( \text{ARTIFACT} \rightarrow (\text{REG\_ART} \rightarrow \text{REG\_ART}) \)

**Pre**: \( art \in \text{REG\_ART} \)

**Post**: \( \text{REG\_ART}' = \text{REG\_ART} \setminus \{art\} \)

(3) Operation to delete groups from the set of registered group agents.

**Name**: del_group\( (old\_grp\_name) \)

**Sig**: \( \text{AGENT} \rightarrow (\text{GRP\_AGS} \rightarrow \text{GRP\_AGS}) \)

**Pre**: \( old\_grp\_name \in \text{REG\_AG} \)

\( \exists \text{grp\_details}_i \in \text{GRP\_AGS} \text{ s.t. } (\text{grp\_details}_i.\text{grp\_name} = old\_grp\_name) \)

**Post**: \( \text{GRP\_AGS}' = \text{GRP\_AGS} \setminus \{\text{grp\_details}_i\} \)

(4) Operation to delete group members from existing group agents.

**Name**: reduce_group\( (old\_grp\_name, old\_grp\_members) \)

**Sig**: \( (\text{AGENT} \times \text{set of AGENT}) \rightarrow (\text{GRP\_AGS} \rightarrow \text{GRP\_AGS}) \)

**Pre**: \( old\_grp\_name \in \text{REG\_AG} \)

\( \forall ag \in old\_grp\_members (ag \in \text{REG\_ART}) \)

\( \exists \text{grp\_details}_i \in \text{GRP\_AGS} \text{ s.t. } (\text{grp\_details}_i.\text{grp\_name} = old\_grp\_name) \)

\( \land (old\_grp\_members \in \text{grp\_details}_i.\text{grp\_members}) \)

\( \text{existing\_members} = \text{grp\_details}_i.\text{grp\_members} \)

**Post**: \( \text{GRP\_AGS}' = \text{GRP\_AGS} \text{ replacing } \text{grp\_details}_i \text{ with } \) \( (old\_grp\_name, \text{existing\_members} \setminus old\_grp\_members) \)
Part III: Approach

(5) Operation to delete contribution relations between agents and artifacts.

<table>
<thead>
<tr>
<th>Name: del_contrib_rel(contrib_rel)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sig: ((\text{AGENT} \times \text{ARTIFACT} \times \text{CAPACITY} \times \text{QUALIFY}) \rightarrow \text{CONTRIB REL} \rightarrow \text{CONTRIB REL}))</td>
</tr>
<tr>
<td>Pre: contrib_rel (\in) CONTRIB_REL</td>
</tr>
<tr>
<td>Post: CONTRIB_REL' = CONTRIB_REL - {contrib_rel}</td>
</tr>
</tbody>
</table>

(6) Operation to delete artifact-based relations between artifacts.

<table>
<thead>
<tr>
<th>Name: del_art_rel(art_rel)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sig: ((\text{ARTIFACT} \times \text{ARTIFACT} \times \text{ARELATION} \times \text{PURPOSE}) \rightarrow \text{ART REL} \rightarrow \text{ART REL}))</td>
</tr>
<tr>
<td>Pre: art_rel (\in) ART_REL</td>
</tr>
<tr>
<td>Post: ART_REL' = ART_REL - {art_rel}</td>
</tr>
</tbody>
</table>

8. Using Contribution Structures

In this section, we describe some of the operations which act on the knowledge of the current state of the contribution structure, the CS model, and so provide the basis for performing various forms of personnel-based RT. A wider coverage of such operations can be found in Appendix B. Here, we first describe some rudimentary operations which can yield true/false type answers and then describe those rudimentary operations which can yield more extensive answers. Based on these rudimentary operations, we outline how it is possible to construct more complex query-related operations which make it possible to answer the frequently asked questions mentioned in Section 2.1. We finish this section by describing the further types of social analyses that become possible as a consequence of the approach. Note that, within this section we focus on those operations that can tell us details about the contributors and their contributions, as opposed to those which solely tell us details about the artifact-based RT relations, examples of the latter being given in Figure 38.
Figure 38: Examples of some rudimentary query operations which can be used to clarify artifact-based RT - (a) do specified artifacts have a common immediate source?; (b) do specified artifacts have a common immediate target?; (c) do specified artifacts have a common intermediate artifact?; (d) is one specified artifact a component of another?; and (e) do specified artifacts have a common containing artifact? Note that it is possible to ask these questions with respect to any of the artifact-based relation types.

8.1. Notes on Notation

In Figure 39, we outline the general form of the query-based operations we will be describing, then provide a few notes on the semi-formal notation used to describe these operations.
query: CS x query_name<query_arguments> --> result

Current state of the model
Name of a query operation which acts on CS
Parameters for the query operation
Result of applying the query to CS

Figure 39: The general form of query operations in our modelling scheme.

It must be noted that, to avoid repetition throughout this section, the checks described below are carried out for all the query operation parameters.

ag_is_registered(ag) ⇒ ag ∈ REG_AG
art_is_registered(art) ⇒ art ∈ REG_ART

8.1.1. Predicates

These operations return true or false responses. Examples read as follows:

(1) The agent ag is a contributor to the artifact art if there is a contribution relation joining ag and art which belongs to the set of registered contribution relations.

ag_contributor_to_art(ag,art) ⇒
contrib_rel(ag,art,_,_) ∈ CONTRIB_REL

(2) All the agents in ag_list are contributors to some artifacts if each agent in this ag_list is a contributor.

all_ags_are_contributors(ag_list) ⇒
∀ ag ∈ ag_list (ag_is_contributor(ag))

(3) The agent ag has relations to other agents, due to their joint collaboration on the same artifacts, if there exists a second agent a such that the intersection of the artifacts that agents ag and a independently contribute to is not empty.

ag_has_relations_to_ags_thro_collab_on_arts(ag) ⇒
∃ a: AGENT s.t. (all_artifacts_ag_contributes_to(ag) ∩ all_artifacts_ag_contributes_to(a)) ≠ ∅
8.1.2. Constructs

These operations return sets of agents, artifacts, or both. Examples read as follows:

(1) Return the set of artifacts for which the predicate $art\_has\_contributor(art)$ returns true.

$$\text{all\_artifacts\_with\_contributors} = \{ art: \text{ARTIFACT} | art\_has\_contributor(art) \}$$

(2) Return the set of agent/artifact pairs for which the predicate $ag\_contributor\_to\_art(ag,art)$ returns true.

$$\text{all\_contribution\_relations} = \{ ag: \text{AGENT}; art: \text{ARTIFACT} | ag\_contributor\_to\_art(ag,art) \}$$

(3) Instantiate the agent $ag$ in the body of the operation with the particular agent $ag$ specified in the operation parameter. Instantiate $art$ with each artifact taken from the set of all registered artifacts in the body of the operation. Return the set of artifacts for which the predicate $ag\_contributor\_to\_art(ag,art)$ returns true.

$$\text{all\_artifacts\_ag\_contributes\_to}(ag) = \{ art\_list: set\ of\ \text{ARTIFACT} | \forall\ art: \text{ARTIFACT} \land art \in art\_list \Rightarrow ag\_contributor\_to\_art(ag,art) \}$$

(4) Instantiate the agent $ag$ in the body of the operation with each agent in $ag\_list$ specified in the operation parameter. Return that resulting subset of agents from $ag\_list$ for which the predicate $ag\_is\_contributor(ag)$ returns true.

$$\text{all\_agents\_in\_ags\_that\_are\_contributors}(ag\_list) = \{ temp\_ag\_list: set\ of\ \text{AGENT} | \forall\ ag \in ag\_list \land ag \in temp\_ag\_list \Rightarrow ag\_is\_contributor(ag) \}$$

8.2. Rudimentary True/False Query Operations

Here, we give a few examples of those queries which provide simple true/false responses. For clarity, we divide these into agent/artifact-centred examples, artifact-centred examples, and agent-centred examples. For each example, we give its signature, followed by its rule.
8.2.1. Agent/Artifact-Centred Examples

Examples of queries based around a specified agent (or agents) and a specified artifact (or artifacts) are given below and are also illustrated in Figure 40 for comprehension:

(1) Is a specified agent a contributor to a specified artifact?

\[ \text{Sig: ag\_contributor\_to\_art}(ag,art) : \text{AGENT} \times \text{ARTIFACT} \rightarrow \text{BOOLEAN} \]

\[ \text{Rule: ag\_contributor\_to\_art}(ag,art) \Rightarrow \text{contrib\_rel}(ag,art,\_\_\_) \in \text{CONTRIB\_REL} \]

(2) Is a specified agent the sole contributor to a specified artifact?

\[ \text{Sig: ag\_sole\_contributor\_to\_art}(ag,art) : \text{AGENT} \times \text{ARTIFACT} \rightarrow \text{BOOLEAN} \]

\[ \text{Rule: ag\_sole\_contributor\_to\_art}(ag,art) \Rightarrow \text{ag\_contributor\_to\_art}(ag,art) \land \neg \exists ag2 : \text{AGENT} \ s.t. \ ((ag \neq ag2) \land \text{ag\_contributor\_to\_art}(ag2,art)) \]

(3) Is a specified agent related to a specified artifact, in some way, through artifact-based relations?
8.2.2. Artifact-Centred Examples

Examples of queries based around a specified artifact (or artifacts) are given below and are also illustrated in Figure 41 for comprehension:

![Artifact-centred examples](image)

**Figure 41:** Artifact-centred examples - (1) artifacts 2 and 3 have full contribution formats, whilst artifact 1 does not; (2) artifacts 2 and 3 are not original contributions as agent Y makes use of agent X’s authored contribution in various ways; (3) artifacts 2 and 3 have the same contributors; (4) artifacts 2 and 3 have identical contribution formats; and (5) artifacts 2 and 3 have the same immediate origin.

(1) Has a full contribution format been provided for a specified artifact?
art \_ has \_ principal(art) \Rightarrow 
contrib\_rel(_,art,Principal,_ ) \in \text{CONTRIB\_REL}

art \_ has \_ author(art) \Rightarrow 
contrib\_rel(_,art,Author,_ ) \in \text{CONTRIB\_REL}

art \_ has \_ documentor(art) \Rightarrow 
contrib\_rel(_,art,Documentor,_ ) \in \text{CONTRIB\_REL}

(2) Is an artifact an original contribution, in that its author does not explicitly make use of other artifacts in its production?

**Sig:** art\_is\_original(art):
\text{ARTIFACT} \rightarrow \text{BOOLEAN}

**Rule:** art\_is\_original(art) \Rightarrow 
\neg \text{art\_references\_artifacts}(art) \land \neg \text{art\_adopts\_artifacts}(art)

art\_references\_artifacts(art) \Rightarrow 
\text{art\_rel}(art,_,References_) \in \text{ART\_REL}

art\_adopts\_artifacts(art) \Rightarrow 
\text{art\_rel}(art,_,Adopts_) \in \text{ART\_REL}

(3) Do two specified artifacts have the same contributors?

**Sig:** arts\_have\_same\_contributors(art1,art2):
\text{ARTIFACT} \times \text{ARTIFACT} \rightarrow \text{BOOLEAN}

**Rule:** arts\_have\_same\_contributors(art1,art2) \Rightarrow 
all\_agents\_that\_contribute\_to\_art(art1) = all\_agents\_that\_contribute\_to\_art(art2)

all\_agents\_that\_contribute\_to\_art(art) =
\{ ag\_list: set of AGENT | \forall ag: AGENT \cdot ag \in ag\_list \Rightarrow 
ag\_contributor\_to\_art(ag,art) \}

(4) Do all specified artifacts have identical contribution formats?

**Sig:** all\_arts\_have\_id\_contribution\_formats(art\_list):
set of\text{ARTIFACT} \rightarrow \text{BOOLEAN}

**Rule:** all\_arts\_have\_id\_contribution\_formats(art\_list) \Rightarrow 
\exists c\_f: set of AGENT \times set of AGENT \times set of AGENT s.t.
\forall art \in art\_list (contrib\_format\_of\_art(art) = c\_f)
contrib_format_of_art(art) =
\{ all_agents_that_contribute_to_art_as_principal(art);
  all_agents_that_contribute_to_art_as_author(art);
  all_agents_that_contribute_to_art_as_documentor(art) \}

all_agents_that_contribute_to_art_as_principal(art) =
\{ ag_list: set of AGENT | \forall ag: AGENT \cdot ag \in ag_list \Rightarrow
  ag_principal_of_art(ag,art) \}

all_agents_that_contribute_to_art_as_author(art) =
\{ ag_list: set of AGENT | \forall ag: AGENT \cdot ag \in ag_list \Rightarrow
  ag_author_of_art(ag,art) \}

all_agents_that_contribute_to_art_as_documentor(art) =
\{ ag_list: set of AGENT | \forall ag: AGENT \cdot ag \in ag_list \Rightarrow
  ag_documentor_of_art(ag,art) \}

ag_principal_of_art(ag,art) \Rightarrow
  contrib_rel(ag,art,Principal,_) \in CONTRIB_REL

ag_author_of_art(ag,art) \Rightarrow
  contrib_rel(ag,art,Author,_) \in CONTRIB_REL

ag_documentor_of_art(ag,art) \Rightarrow
  contrib_rel(ag,art,Documentor,_) \in CONTRIB_REL

(5) Do all specified artifacts have an identical immediate origin?

\textbf{Sig:} \_\_arts\_have\_identical\_immed\_origin(art_list):
\_\_set of ARTIFACT \rightarrow \_\_BOOLEAN

\textbf{Rule:} \_\_arts\_have\_identical\_immed\_origin(art_list) \Rightarrow \exists art \_\_ARTIFACT s. t.
\_\_

\forall art \in art_list (art1 \in \_\_arts\_related\_to\_art\_thro\_in\_relations(art))

\_\_arts\_related\_to\_art\_thro\_in\_relations(art) =
\{ art_list: set of ARTIFACT | \forall a: ARTIFACT \cdot a \in art_list \Rightarrow
  art1\_related\_to\_art2\_thro\_in\_relations(art,a) \}

art1\_related\_to\_art2\_thro\_in\_relations(art1,art2) \Rightarrow
  art\_rel(art2,art1,_,__) \in ART\_REL

8.2.3. Agent-Centred Examples

Examples of queries based around a specified agent (or agents) are given below and are also illustrated in Figure 42 for comprehension:
Figure 42: Agent-centred examples - (1) agent Z is a group, so agent C only contributes in a group capacity; (2) agent Z has contributed in all PAD capacities, so has its group members, but agent X has not; (3) agents A and B are related through group membership and collaboration, whilst agents X and B are only related through collaboration; (4) agents A and X are not directly related, but agent B is an intermediary agent, as agents A and B are related through group membership and collaboration, and agents X and B are related through collaboration; and (5) agents A, B and C do not contribute to identical artifacts, although artifact 1 is common to all.

(1) Has a specified agent contributed to artifacts in an individual capacity?

Sig: ag_is_contributor_in_individual_capacity(ag):
AGENT \rightarrow BOOLEAN

Rule: ag_is_contributor_in_individual_capacity(ag) ⇒
ag_is_contributor(ag) \land ag_is_individual(ag)

ag_is_contributor(ag) ⇒
contrib_rel(ag,_,_,_,_,_) ∈ CONTRIBUTION

ag_is_individual(ag) ⇒
\neg ag_is_group(ag)

ag_is_group(ag) ⇒
grp_details(ag,_,_) ∈ GROUP_AGG

(2) Has a specified agent contributed in all capacities at some time within a project?

Sig: ag_contributed_in_all_capacities(ag):
AGENT \rightarrow BOOLEAN

Rule: ag_contributed_in_all_capacities(ag) ⇒
ag_is_principal(ag) \land ag_is_author(ag) \land ag_is_documentor(ag)
Part III: Approach

(3) Are two specified agents related?

**Sig:** $\text{ags\_related}(ag_1, ag_2)$:

$\text{AGENT} \times \text{AGENT} \rightarrow \text{BOOLEAN}$

**Rule:**

$\text{ags\_related}(ag_1, ag_2) \Rightarrow$

- $\text{ags\_related\_thro\_collab\_on\_artifacts}(ag_1, ag_2)$
- $\text{ags\_related\_thro\_group\_membership}(ag_1, ag_2)$
- $\text{ags\_related\_thro\_artifact\_dependencies}(ag_1, ag_2)$

$\text{ags\_related\_thro\_collab\_on\_artifacts}(ag_1, ag_2) \Rightarrow$

$(\text{all\_artifacts\_ag\_contributes\_to}(ag_1) \cap \text{all\_artifacts\_ag\_contributes\_to}(ag_2)) \neq \emptyset$

$\exists a: \text{AGENT s.t. ag\_is\_group}(a) \land ag_1\_group\_and\_ag_2\_member(a, ag_1) \land ag_1\_group\_and\_ag_2\_member(a, ag_2)$

$\text{ags\_related\_thro\_artifact\_dependencies}(ag_1, ag_2) \Rightarrow$

$\exists \text{art}_1, \text{art}_2: \text{ARTIFACT s.t. art}_1 \in \text{all\_artifacts\_ag\_contributes\_to}(ag_1) \land art_2 \in \text{all\_artifacts\_ag\_contributes\_to}(ag_2) \land art_1\_related\_to\_art_2\_thro\_relations(art_1, art_2)$

(4) Do two specified agents have a mediating agent if they are not directly related themselves?

**Sig:** $\text{ags\_have\_mediating\_agent}(ag_1, ag_2)$:

$\text{AGENT} \times \text{AGENT} \rightarrow \text{BOOLEAN}$

**Rule:**

$\neg \text{ags\_have\_mediating\_agent}(ag_1, ag_2) \Rightarrow$

$\exists ag_3: \text{AGENT s.t. (ags\_related}(ag_1, ag_3) \land \text{ags\_related}(ag_2, ag_3))$
Do all the agents in a specified list of agents contribute to identical artifacts?

**Sig:** all_aggs_contribute_to_id_artifacts (ag_list):
set ofAGENT → BOOLEAN

**Rule:** all_aggs_contribute_to_id_artifacts (ag_list) ⇒
∃ arts: set ofARTIFACT s.t.
∀ ag ∈ ag_list (all_artifacts_ag_contributes_to(ag) = arts)

8.3. Rudimentary Construct Query Operations

Here, we give a few examples of those queries which provide more detailed responses to identify agents, artifacts, or both, in accordance with specified criteria. Again, and for clarity, we divide these into agent/artifact-centred examples, artifact-centred examples, and agent-centred examples. For each example, we give its signature, followed by its rule.

8.3.1. Agent/Artifact-Centred Examples

Examples of queries based around a specified agent (or agents) and a specified artifact (or artifacts) are given below and are also illustrated in Figure 43 for comprehension:

**Figure 43:** Further agent/artifact-centred examples - (1) agent X has contributed to artifact 1 as principal, whilst agent Y has contributed to artifact 1 as author and documentor; (2) agent X is assumed to have approved artifact 1; and (3) agent X is the sponsor of artifact 1, whilst agent Y is its representative, and agent X is the true author of artifact 2.
(1) How has a specified agent contributed to a specified artifact?

\[
\text{Sig: } \text{contribution\_type}(ag, art): \quad \text{AGENT} \times \text{ARTIFACT} \rightarrow \text{set of CAPACITY}
\]

\[
\text{Rule: } \text{contribution\_type}(ag, art) = \\
\{ c\_list: \text{set of CAPACITY} | \\
\text{Principal} \in c\_list \Rightarrow \text{ag\_principal\_of\_art}(ag, art) \land \\
\text{Author} \in c\_list \Rightarrow \text{ag\_author\_of\_art}(ag, art) \land \\
\text{Documentor} \in c\_list \Rightarrow \text{ag\_documentor\_of\_art}(ag, art) \}
\]

(2) What is the status of a specified agent's contribution as Principal to a specified artifact?

\[
\text{Sig: } \text{principal\_status\_type}(ag, art): \quad \text{AGENT} \times \text{ARTIFACT} \rightarrow \text{PQUALIF}
\]

\[
\text{Rule: } \text{principal\_status\_type}(ag, art) = \\
\{ p\_qualif: \text{PQUALIF} | \\
\text{Approved} \Rightarrow \text{art\_approved\_by\_ag}(ag, art) \lor \\
\text{PendingApproval} \Rightarrow \text{art\_pending\_by\_ag}(ag, art) \lor \\
\text{NotApproved} \Rightarrow \text{art\_not\_approved\_by\_ag}(ag, art) \}
\]

\[
\text{art\_approved\_by\_ag}(ag, art) \Rightarrow \\
\text{contrib\_rel}(ag, art, \_, \_\text{Approved}) \in \text{CONTRIB REL}
\]

\[
\text{art\_pending\_by\_ag}(ag, art) \Rightarrow \\
\text{contrib\_rel}(ag, art, \_, \_\text{PendingApproval}) \in \text{CONTRIB REL}
\]

\[
\text{art\_not\_approved\_by\_ag}(ag, art) \Rightarrow \\
\text{contrib\_rel}(ag, art, \_, \_\text{NotApproved}) \in \text{CONTRIB REL}
\]

(3) In what social role has a specified agent contributed to a specified artifact?
**Sig:** social\_role\_type(ag,art):

\[
\text{AGENT} \times \text{ARTIFACT} \rightarrow \text{set of SROLE}
\]

**Rule:** social\_role\_type(ag,art) =

\[
\{s\_roles: \text{set of SROLE}\}
\]

TrueAuthor ∈ s\_roles ⇒ ag\_true\_author\_art(ag,art) ∧

Devisor ∈ s\_roles ⇒ ag\_devisor\_art(ag,art) ∧

Relayer ∈ s\_roles ⇒ ag\_relayer\_art(ag,art) ∧

Sponsor ∈ s\_roles ⇒ ag\_sponsor\_art(ag,art) ∧

Representative ∈ s\_roles ⇒ ag\_represent\_art(ag,art) ∧

NominalAuthor ∈ s\_roles ⇒ ag\_nom\_author\_art(ag,art) ∧

GhostAuthor ∈ s\_roles ⇒ ag\_ghost\_author\_art(ag,art)

\[
\text{ag\_true\_author\_art}(ag,art) \Rightarrow
\]

\[
\text{ag\_principal\_of\_art}(ag,art) \land \text{ag\_author\_of\_art}(ag,art) \land
\]

\[
\text{ag\_documentor\_of\_art}(ag,art)
\]

\[
\text{ag\_devisor\_art}(ag,art) \Rightarrow
\]

\[
\text{ag\_principal\_of\_art}(ag,art) \land \text{ag\_author\_of\_art}(ag,art)
\land
\]

\[
\text{~ag\_documentor\_of\_art}(ag,art)
\]

\[
\text{ag\_relayer\_art}(ag,art) \Rightarrow
\]

\[
\text{~ag\_principal\_of\_art}(ag,art) \land \text{~ag\_author\_of\_art}(ag,art)
\land
\]

\[
\text{ag\_documentor\_of\_art}(ag,art)
\]

\[
\text{ag\_sponsor\_art}(ag,art) \Rightarrow
\]

\[
\text{ag\_principal\_of\_art}(ag,art) \land \text{~ag\_author\_of\_art}(ag,art)
\land
\]

\[
\text{~ag\_documentor\_of\_art}(ag,art)
\]

\[
\text{ag\_represent\_art}(ag,art) \Rightarrow
\]

\[
\text{~ag\_principal\_of\_art}(ag,art) \land \text{ag\_author\_of\_art}(ag,art)
\land
\]

\[
\text{ag\_documentor\_of\_art}(ag,art)
\]

\[
\text{ag\_nom\_author\_art}(ag,art) \Rightarrow
\]

\[
\text{ag\_principal\_of\_art}(ag,art) \land \text{~ag\_author\_of\_art}(ag,art)
\land
\]

\[
\text{ag\_documentor\_of\_art}(ag,art)
\]

\[
\text{ag\_ghost\_author\_art}(ag,art) \Rightarrow
\]

\[
\text{~ag\_principal\_of\_art}(ag,art) \land \text{ag\_author\_of\_art}(ag,art)
\land
\]

\[
\text{~ag\_documentor\_of\_art}(ag,art)
\]
8.3.2. Artifact-Centred Examples

Examples of queries based around a specified artifact (or artifact) are given below and are also illustrated in Figure 44 for comprehension:

![Diagram of artifact relations]

**Figure 44**: Further artifact-centred examples - (1) agents X and Y are the contributors to artifact 1; (2) agents Y and Z reference artifact 1 in artifact 2, as principal and as author and documentor respectively, and agent Y references artifact 1 in artifact 3 as principal, author and documentor; (3) the authorial status of agent Y, the author of artifact 1, is creator, whilst the authorial status of the authors of artifact 2 and artifact 3, agent Z and agent Y respectively, is referencer; (4) out of artifacts 1, 2 and 3, agent Y has contributed to all of them; and (5) out of artifacts 1, 2 and 3, artifacts 1 and 3 have identical documentors, namely agent Y.

1. Who are the contributors to a specified artifact?

   **Sig**: all_agents_that_contribute_to_art(art):
   
   ARTIFACT → set of AGENT

   **Rule**: all_agents_that_contribute_to_art(art) =
   
   \{ ag_list: set of AGENT | \forall ag: AGENT \cdot ag \in ag_list \Rightarrow
g ag_contributor_to_art(\text{ag}, art)\}

2. Which agents reference a specified artifact in their contributions?

   **Sig**: all_agents_referencing_art(art):
   
   ARTIFACT → set of AGENT

   **Rule**: all_agents_referencing_art(art) =
   
   \{ ag_list: set of AGENT | \forall ag: AGENT \cdot ag \in ag_list \Rightarrow
   art_referenced_by_ag(\text{ag}, \text{art})\}

   art_referenced_by_ag(\text{ag}, \text{art}) \Rightarrow
   \exists a: ARTIFACT s.t. art_rel(a, art, References, _) \in ART_REL \wedge
   contrib_rel(\text{ag}, a, _, _) \in CONTRIB_REL
(3) What is the authorial status of a specified artifact's author contributor?

Sig: authorial_status_of_art(art):
    ARTIFACT → set of AQUALIF
Rule: authorial_status_of_art(art) =
    \{ qualif_list: set of AQUALIF | 
        (Referencer ∈ qualif_list ⇒ art_refs_artifacts(art) ∧ 
        Adopter ∈ qualif_list ⇒ art_adopts_artifacts(art)) ∨ 
        Creator ∈ qualif_list ⇒ art_is_original(art) \}

art_refs_artifacts(art) ⇒
    art_rel(_, _, References _) ∈ ART_REL

art_adopts_artifacts(art) ⇒
    art_rel(_, _, Adopts _) ∈ ART_REL

art_is_original(art) ⇒
    ¬ art_references_artifacts(art) ∧ ¬ art_adopts_artifacts(art)

(4) Which agents have contributed to all of the artifacts in a specified list of artifacts?

Sig: all_agents_that_contribute_to_all_arts(art_list):
    set of ARTIFACT → set of AGENT
Rule: all_agents_that_contribute_to_all_arts(art_list) =
    \{ ag_list: set of AGENT | ∀ ag: AGENT • ag ∈ ag_list ⇒ 
        ∀ art ∈ art_list s.t. ag_contributor_to_art(ag, art) \}

(5) Which artifacts have identical documentors amongst a specified list of artifacts?

Sig: all_arts_in_arts_that_have_id_documentors(art_list):
    set of ARTIFACT → set of ARTIFACT
Rule: all_arts_in_arts_that_have_id_documentors(art_list) =
    \{ ∀ art1, art2 ∈ all_arts_in_arts_that_have_documentors(art_list) | 
        all_ags_that_contribute_to_art_as_documentor(art1) ∩ 
        all_ags_that_contribute_to_art_as_documentor(art2) ≠ ∅ \}

all_arts_in_arts_that_have_documentors(art_list) =
    \{ temp_art_list: set of ARTIFACT | ∀ art ∈ art_list • art ∈ temp_art_list ⇒ 
        art_has_documentor(art) \}

art_has_documentor(art) ⇒
    contrib_rel(_, Documentor, _) ∈ CONTRIB_REL
8.3.3. Agent-Centred Examples

Examples of queries based around a specified agent (or agents) are given below and are also illustrated in Figure 45 for comprehension:

Figure 45: Further agent-centred examples, reusing the previous diagram - (1) agent Y has contributed to artifacts 1, 2 and 3; (2) agent Y is the principal of artifacts 2 and 3; (3) agents X and Z are related through an artifact-based relation between artifacts 1 and 2, to which they respectively contribute; (4) agents X, Y and Z contribute to artifacts 1, 2 and 3; and (5) out of agents X, Y and Z, agents X and Y are related through collaboration on artifact 1, and agents Y and Z are related through collaboration on artifact 2.

(1) Which artifacts has a specified agent contributed to?

Sig: all _arts _ag_contributes _to(ag):
    AGENT → set of ARTIFACTS
Rule: all _arts _ag_contributes _to(ag) =
    { art_list: set of ARTIFACT | ∀ art: ARTIFACT • art ∈ art_list ⇒
      ag_contributor_to_art(ag,art) }
(2) Which artifacts are a specified agent the principal of?

```
Sig: all_artifacts_ag_contributes_to_as_principal(ag):
    AGENT \rightarrow set of ARTIFACT
Rule: all_artifacts_ag_contributes_to_as_principal(ag) =
    \{ art_list: set of ARTIFACT | \forall art: ARTIFACT • art \in art_list \Rightarrow
      ag_principal_of_art(ag,art) \}

ag_principal_of_art(ag,art) \Rightarrow
  contrib_rel(ag,art,Principal,_) \in CONTRIB_REL
```

(3) Which other agents are related to a specified agent through artifact-based relations?

```
Sig: all_ags_ag_related_to_thro_art_dependencies(ag):
    AGENT \rightarrow set of AGENT
Rule: all_ags_ag_related_to_thro_art_dependencies(ag) =
    \{ ag_list: set of AGENT | \forall a: AGENT • a \in ag_list \Rightarrow
      ags_related_thro_art_dependencies(ag,a) \}

ags_related_thro_art_dependencies(ag1,ag2) \Rightarrow
  \exists art1 art2: ARTIFACT s.t. art1 \in all_artifacts_ag_contributes_to(ag1) \land
  art2 \in all_artifacts_ag_contributes_to(ag2) \land
  art1_related_to_art2_thro_relations(art1,art2)
```

(4) Which artifacts do a specified list of agents contribute to?

```
Sig: all_artifacts_ags_contribute_to(ag_list):
    set of AGENT \rightarrow set of ARTIFACT
Rule: all_artifacts_ags_contribute_to(ag_list) =
    \{ art_list: set of ARTIFACT | \forall ag \in ag_list; art: ARTIFACT • art \in art_list \Rightarrow
      ag_contributor_to_art(ag,art) \}
```

(5) Which agents amongst a specified list of agents are related through collaboration on the same artifact?

```
Sig: all_ags_in_ags_related_thro_collab_on_arts(ag_list):
    set of AGENT \rightarrow set of AGENT
Rule: all_ags_in_ags_related_thro_collab_on_arts(ag_list) =
    \{ temp_ag_list: set of AGENT | \forall ag1 \in ag_list; ag2 \in ag_list •
      ag1 \in temp_ag_list \Rightarrow ags_related_thro_collab_on_arts(ag1,ag2) \}
```
8.4. Towards More Complex Query Operations

Using the above rudimentary operations as the basic building blocks, we now show how it is possible to answer more complex questions. However, we do not go into the same detail here, only providing some high-level examples below:

1. Return all those artifacts which have agent ag1 as their divisor and agent ag2 as their relayer.

\[
\text{result} = \text{all\_artifacts\_ag\_is\_devisor\_of}(ag1) \cap \text{all\_artifacts\_ag\_is\_relayer\_of}(ag2)
\]

2. Return whether or not the same agent(s) have been responsible for approving all the artifacts which have been approved on a project.

\[
\text{result}_1 = \text{all\_artifacts\_that\_have\_been\_approved}
\]
\[
\text{final\_result} = \text{all\_arts\_with\_identical\_principals}(\text{result}_1)
\]

3. Return all those artifacts documented by those agents who are related to, though not the same as, those agents who were the principals of artifact art.

\[
\text{result}_1 = \text{all\_agents\_that\_contribute\_to\_art\_as\_principal}(art)
\]
\[
\text{result}_2 = \text{all\_external\_related\_agents\_to\_ags}(\text{result}_1)
\]
\[
\text{final\_result} = \text{all\_arts\_ags\_contribute\_to\_as\_documentor}(\text{result}_2)
\]

8.5. Some Answers to Frequently Asked Questions

Rather than exhaustively determining the more complex types of question that can be answered about those involved in the RE process, we return to the frequently asked questions of Section 2.1. There, we listed some examples of the kinds of question which, if left unanswered, were found to result in practitioner claims to have experienced RT problems. The last five questions in this list, repeated below, are relatively straightforward and can be readily answered using simple operations...
like those given in the above sub-sections:

(6) "To whom should I refer for more information about this requirement?"

(7) "Within the remit of which group do decisions about this requirement lie?"

(8) "Who was responsible for copying this piece of information into this requirement?"

(9) "Who uses this requirement and how?"

(10) "On which requirements do certain agents collaborate together and how?"

However, the first five questions in this list are slightly more involved, so we briefly describe how these can also be readily addressed in turn.

8.5.1. FAQ 1: Involvement

Box 15: Question 1 of the frequently asked questions - Involvement.

"Who has been involved in the production of this requirement and how?"

To identify all those agents who have been involved in the production of a requirement, we would need to trace the full chain of contribution behind its production process. This can be done using the artifact-based RT relations to trace the requirement back to its origin and by extracting the corresponding contribution formats along the way. The ultimate origin of a requirement would be reached when it is not possible to trace back to any other artifacts through further artifact-based RT relations, though a more relaxed origin would be one which has no further outgoing adopts relations, although it might reference other artifacts for background information and the like. Note that a requirement may have multiple origins, so it would be essential to trace those involved in all the contributing branches. Dependent on the level of specificity required, we can carry out this personnel-based RT at various levels of detail, like selectively tracing the chain of principals, authors, or documentors involved in the production of a requirement.

8.5.2. FAQ 2: Responsibility

Box 16: Question 2 of the frequently asked questions - Responsibility.

"Who was originally responsible for this requirement, who is currently responsible for it, and at what points in its life has this responsibility changed hands?"

To identify the agent(s) originally responsible for a requirement, we would again need to use the
artifact-based RT relations to trace the requirement back to its origin(s), and so extract the original contribution format(s). Again, a more relaxed notion of origin is likely to be what is required. Likewise, we can find the agent(s) currently responsible for a requirement by using the artifact-based RT relations to trace the requirement to its target(s), then extracting the corresponding contribution format(s). The ultimate target of a requirement would be reached when it is not possible to trace forward to any other artifacts through further artifact-based RT relations, though a more relaxed target would be one which is not directly adopted elsewhere, although it might be referenced in other artifacts. Dependent on the type of responsibility we are interested in, which could crudely be for the appearance, content, or effect of the requirement, we can focus on retrieving the appropriate capacities from the contribution format in the personnel-based traces. By retrieving all the various artifact-based traces between the origin(s) and the target(s), and extracting details of those contributing along these chains, it is possible to analyse how, when, and on what artifacts the agents assuming various types of responsibility for a requirement changes.

8.5.3. FAQ 3: Working Arrangement

"At what points in this requirement's life have the working arrangements of all those involved been changed?"

**Box 17: Question 3 of the frequently asked questions - Working Arrangement.**

To identify changes or trends in the working arrangements of agents throughout a requirement's life, by which we mean the alignments of those that jointly contribute to the artifacts concerned, again we would need to make use of the full chains of contribution behind a requirement. This question can be answered, not only by examining changes with respect to those involved in the specific capacities of the contribution format for each artifact in the chain, but more finely by highlighting the changes in the social roles and role relations of those contributing.

8.5.4. FAQ 4: Change Notification

"Who needs to be involved in, or informed of, any changes proposed to this requirement?"

**Box 18: Question 4 of the frequently asked questions - Change Notification.**

To identify those to involve in and inform about changes, we would need to use the forwards and backwards artifact-based RT relations to identify all the artifacts impacted by a proposed change in both directions, and then go on to identify the corresponding contributors potentially affected. If we know more about the nature of the change, and the nature of the artifact-based RT relations, these details can be used to filter those most likely to be affected by the change and also to
prioritise those to involve and inform.

8.5.5. FAQ 5: Ramification

"What are the ramifications, with respect to the loss of possible requirements-related knowledge, if a specific individual or group leaves a project?"

Box 19: Question 5 of the frequently asked questions - Ramification.

To answer this question, we would need to identify all of an agent's contributions, the capacities in which they have contributed, and their social relations with respect to other agents. Using such information, it would be possible to assess whether there are suitable agents which have directly collaborated with the agent in question and likely to possess adequate experiential knowledge to that which may be lost, else assess the most suitable alternative agent who could act as possible stand-in and take over the specified agent's commitments. Through the artifact-based RT relations, we could also identify those agents who's contributions the agent in question has depended upon, possibly revealing necessary social contacts for the replacement agent. In the same way, we could identify those agents which make use of the agent in question's contributions, so identify those agents that may have outstanding requirements for communication and future contact with the replacement agent. With visibility of all this information, we would be in a position to assess the amount of up-heaval that would be caused, and so to determine how best to smooth-over the change process. We would also be in a position to identify those areas where a replacement agent would need de-briefing.

8.6. Towards Further Analyses

A by-product of the approach is that it generates relational data which can be used to describe and analyse richer details about the wider social structure underlying the RE process. For instance, the data gathered about agents, artifacts, and their various interrelations, can be organised in such a way as to carry out numerous forms of social network analyses as described by [Scott 1991]. Such analyses would require the data to be described in terms of incidence and adjacency matrices, as illustrated in Tables 14 to 16. The former type of matrix could be used to indicate the number of agents in common between pairs of artifacts or to list the agents who have contributed to pairs of artifacts. They could also be used to indicate the number of artifacts which have been contributed by both agents jointly or to list such artifacts. The latter type of matrix could be used to indicate information like the capacities or roles in which agents have contributed to artifacts.


<table>
<thead>
<tr>
<th>Artifact 1</th>
<th>Artifact 2</th>
<th>Artifact 3</th>
<th>Artifact 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artifact 1</td>
<td>-</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Artifact 2</td>
<td>1</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>Artifact 3</td>
<td>1</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>Artifact 4</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**Table 14:** An incidence matrix indicating the number of relations between artifacts. It could be used to signify the different types of these relations.

<table>
<thead>
<tr>
<th>Agent 1</th>
<th>Agent 2</th>
<th>Agent 3</th>
<th>Agent 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent 1</td>
<td>-</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Agent 2</td>
<td>2</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td>Agent 3</td>
<td>1</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>Agent 4</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**Table 15:** An incidence matrix indicating the number of relations between agents. It could be used to signify the different types of these relations.

<table>
<thead>
<tr>
<th>Artifact 1</th>
<th>Agent 1</th>
<th>Agent 2</th>
<th>Agent 3</th>
<th>Agent 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artifact 1</td>
<td>A</td>
<td>D</td>
<td>-</td>
<td>P</td>
</tr>
<tr>
<td>Artifact 2</td>
<td>-</td>
<td>D</td>
<td>D</td>
<td>PA</td>
</tr>
<tr>
<td>Artifact 3</td>
<td>PAD</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Artifact 4</td>
<td>P</td>
<td>AD</td>
<td>-</td>
<td>P</td>
</tr>
</tbody>
</table>

**Table 16:** An adjacency matrix indicating contribution relations between agents and artifacts. It could be used to signify the different capacities in which they have contributed.

Many forms of graphical and statistical analyses could be performed upon such rudimentary data, a summary of some of the information they could uncover being listed below:

1. Information about the nature of the relations between agents. For example, their transience, durability, frequency, intensity, directionality, mutuality, reciprocity, and strength. Such information could be used to inform whether dependencies between agents are strong, weak, essential, desirable, bi-directional, uni-directional, and so forth.

2. Information about the direct and indirect relations between agents. Such information could be used to identify which agents possibly communicate with each other, either directly or through chains of intermediaries, and those agents amongst whom resources and information
are able to flow freely and easily. Using graph theoretic notions, such as paths and walks, measures such as the distances between agents, their reachability, and ease of contact could be determined.

(3) Information about the texture of the social network. This is because the extent to which all possible relations are present amongst the agents in a project indicates the level of linkage or cohesion amongst them. Such information could be used to provide measures of density, completeness, connectedness, and inclusiveness amongst the agents. Such information could also be used to examine aspects like the total contributions of a specific agent, say as a percentage of the total contributions of all the agents in a project, and so gauge the extent of an agent's involvement in a project relative to other agents.

(4) Information about both the global and local centres of attention and popularity. Such information could be used to indicate those agents in prominent, strategic, powerful, and significant positions in a project. This is important information because the provision of requirements-related information by agents in a project is likely to depend, not only on the motivation of those agents with the information, but also on the strategic location of an agent's contacts in the overall information flow. This has been referred to elsewhere as the information diffusion model [Granovetter 1974].

(5) Information about the personal networks anchored around specific agents. Such information could be used to identify various types of agent. For instance: (a) those agents with fewer and weaker connections as being peripheral agents; (b) those agents with more and stronger connections as being central agents; (c) those agents which are more central than the other agents to which they are connected as being peak agents; (d) those central agents connecting two or more agents as being bridge agents; and (e) those agents not connected to any others as being isolated agents.

(6) Information about the various types of social structure and cohesive agent sub-groupings that have been formed. Such information could be used to identify: (a) those mutually connected agents as agent cliques; (b) those agents which are the direct contacts of a peak agent as agent clusters; (c) those agents participating together most frequently as agent cores; (d) those agents which sometimes participate with the core, but never as a group, as agent primary circles; and (e) those agents who infrequently participate as agent secondary circles. In turn, such information could inform about aspects like cooperation and social pressures within a project.

On a related note, Biddle and Thomas explain how the relations between agents can provide sociometric data, the relations between agents and their roles can provide information about worker specifications, and the relations between agent roles can provide information about actual
Part III: Approach

working arrangements [Biddle & Thomas 1966]. With data such as this, it could then be possible to make comparisons between any formal and predefined organisation structures and the information provided about the informal and actual working practices. Does the contribution structure underlying a particular project mimic the formal structure of the organisation conducting the project? This would be a useful facility for project management as it would provide a more realistic picture of the working organisation and inform process improvement. Furthermore, and with information gathered across projects, it could be possible to examine whether the different contribution structures underlying projects have different effects on the overall quality of the artifacts produced.

9. Summary of Part III

In Part III, we have provided a detailed description of an approach to address the crux of the RT problem. We have shown how this approach makes relevant details about the social structure that gave rise to the tangible artifacts produced in RE explicit and so makes personnel-based RT possible. We have outlined some frequently asked questions that are problematic to answer and have described why current practice leads to such difficulties. We have listed the fundamental requirements for an approach to tackle this more central issue and have highlighted some political issues that would need to be considered when introducing any such an approach into an organisation. We have detailed each of the steps of the approach, summarised the foundations upon which they are based, and illustrated how they operate through examples. In so doing, we have demonstrated how it is possible to build an increasingly detailed picture of the contribution structure underlying RE from minimal extra input. We have further provided a model-based specification of the approach and its operation to clarify how contribution structures can be modelled and used. Finally, we have demonstrated how the approach provides the means through which we are able to address the frequently asked questions about those involved in the RE process, and have further pointed out the many extra opportunities it provides in terms of value-added information.
Part IV: Validation

1. Overview of Part IV

In Part IV, we describe how we validated whether the approach presented in Part III of this thesis addresses the crux of the RT problem identified in Part II. We first outline our method of evaluation, then explain how this involved the development of a prototype tool in which the approach could be implemented. Based on the earlier description of what the providers and end-users of pre-RT require, and on the specification of how the approach operates, we list the main requirements to be met by tool support for modelling and using contribution structures. We also summarise some of the assumptions and decisions that underlie the resulting tool. We follow this by a description of the main features of the tool, along with details as to some considerations for implementing a production version. We then use a scenario to illustrate how the approach can be operationalised in the tool, to show how it provides the ability to extend conventional forms of artifact-based RT with information about the contribution structure, and to outline the insight that can be brought to the scenario. We then report on a case study we conducted. This applied the approach to real data obtained from within an industrial setting. We describe this study in detail, discuss its results, and finally end with a critical evaluation.

2. Evaluation Method

For the purpose of this thesis, we only intend to assess whether the approach we have described is feasible in practice, and whether it is able to effectively identify the kinds of detail about those involved in the RE process that practitioners have been found to require. We do not intend to prove conclusively the generalisability and scaleability of the approach. We do not intend to assess whether we have the optimum scheme for delineating the capacities or social roles in which agents contribute. We also do not intend to assess whether we have a complete set of artifact-based RT relations. All such aspects are discussed in the conclusions of this thesis. Instead, and so as to provide proof of concept, our evaluation method consists of a number of levels through which the adequacy of the approach is to be assessed:

(1) Providing a model-based specification of the approach through which to demonstrate the
feasibility of the underlying concepts in principle. (Previously given in Part III.)

(2) Developing an RS to outline what is needed to implement the approach. (See Section 3.)

(3) Constructing a prototype tool to illustrate how these requirements can be met in a practical way. (See Section 4.)

(4) Giving a small scenario to describe the potential of the approach, followed by a larger case study of the application of the approach in a real setting, so as to be in a position to discuss the advantages and limitations of the approach compared to current best practice. (See Sections 5 and 6.)

(5) Critically evaluating the approach based on all of the above. (See Section 7.)

Our evaluation method therefore makes use of subjective techniques which yield qualitative data. This is in contrast to the empirical forms of evaluation, common in areas like human-computer interaction, which use techniques such as system monitoring and logging, or psychologically-based experiments to yield quantitative data. How to use such forms of experimentation in the area of RE, particularly when many techniques and tools only reap their benefits through longitudinal use from the outset of a project, is currently the subject of much interest [Ryan 1995]. We recognise that a better evaluation could be carried out if we were able to shadow a real project in which conventional artifact-based RT were implemented with one in which contribution structures were also modelled and used for RT purposes. In this way, we would be able to compare pertinent factors across the two projects over time. However, getting access to such a project at its onset, and maintaining this access throughout its life, is problematic, costly, and beyond the scope of this thesis.

3. Tool Support

In the world of software engineering: "A software prototype is a partial implementation of a system constructed primarily to enable customers, users, or developers, to learn more about a problem or its solution" [Davis 1990]. In this section, as well as in the next, we indicate how we used prototyping to demonstrate, refine, and evaluate our approach. Our prototyping strategy is summarised in Figure 46. We first describe how the prototype tool was constructed in a disciplined manner. We mention the objectives for the prototype, delimit its scope of concern, and then outline its high-level requirements. Here, we emphasise those user requirements for the providers and end-users that are essential to operationalise the approach, then go on to mention the main assumptions and design decisions underlying the resulting tool. The architecture of the prototype itself is described in Section 4.
3.1. Objectives and Scope

The main objective for prototyping was to construct a vehicle through which we could demonstrate, refine, and evaluate our approach. For this reason, the scope of the prototype tool was restricted to focus on the novel concepts we are proposing. The prototype was used in an explanatory capacity, to determine how conventional notions of artifact-based RT could be extended with associated contribution structures, and to demonstrate the various forms of personnel-based RT that this could make possible. It was also used in an exploratory and experimental capacity, to iteratively refine the approach and the tool based on a clearer understanding of the operational issues involved.
3.2. High-Level Requirements

To outline the high-level requirements necessary to support the approach in practice, we draw together the key requirements for pre-RT given in Part II of this thesis, along with the more specific requirements for modelling and using contribution structures. The full RS for pre-RT is included in Appendix C.1. and the specialised RS for defining, maintaining, and using contribution structures is given in Appendix C.2. In summary these requirements are:

1. **Obtain, record, organise, and maintain details about artifacts and artifact-based RT relations.** A way must be provided to get the heterogeneous artifacts produced in the RE process into the system. The system must be able to deal with the temporal, developmental, and auxiliary relations which exist within and between the artifacts.

2. **Obtain, record, organise, and maintain details about agents and predefined organisational relations.** A way must be provided to get the details required about the agents involved in the RE process into the system. The system must be able to deal with any predefined relations which exist between agents to define various organisational structures.

3. **Obtain, record, organise, and maintain details about contribution relations.** A way must be provided to get details regarding the agents who have contributed to an artifact, and in what capacity, into the system.

4. **Model contribution structure.** The system must be able to use the above information to define and maintain an up-to-date model of the contribution structure. The system must be able to do this as either the contributions, contributors, or their various interrelations evolve and become more complex over time.

5. **Use contribution structure.** The system must be able to use the evolving model of the contribution structure to support the required traceability of those involved in the RE process during any stage of the development process.

3.3. Task Model

Using a task model, we describe those particular requirements necessary to operationalise the approach. As task analysis has received much coverage in the human-computer interaction literature [Diaper 1989, Johnson 1992, Johnson et al. 1984], it is not our intention to go into the various approaches and their underlying assumptions here. Instead, when describing our model, we have chosen to reflect some of the basic conventions and terminology found in the Task Knowledge Structures framework for task modelling [Johnson & Johnson 1991, Johnson et al. 1988]. This is because the approach focuses on modelling work tasks and the models themselves are amenable to the kind of decomposition which is well suited to subsequent prototyping
activities. It is important to point out that, as contribution structures are not currently provided for and used in RE, there is no extant system and no extant task performers with which to carry out task analysis. However, we still believe it is both possible and useful to adopt a task analytic approach to inform how those practices which do not presently exist would need to be supported.

3.3.1. A Role-Oriented Task Model

To focus on demonstrating the basic concepts of the approach in the prototype tool, the task model considers what a single provider needs to do to instantiate the approach and what a single end-user needs to do to invoke personnel-based RT. Based on their distinct work tasks, we therefore describe the task model for modelling and using contribution structures in a role-oriented manner. This model serves to describe the main flow of events, like those shown in Figure 47, and was further used to drive the design of the tool. Although it is important to note that these two roles can overlap, and can even encompass more specific roles, this has been temporarily ignored for now. Issues to do with collaborative and distributed working have also been ignored for now.
In the following sub-sections, we summarise the high-level details of this role-oriented task model. The concepts we use to describe the internal components of the tasks are first described in Box 20. A more comprehensive model, one which goes on to identify the fundamental actions and objects of the task procedures, can be found in Appendix C.3.
**Goal:** The top-level state of affairs that a particular task can produce.

**Sub-goal:** A decomposition of a goal which delineates a part that can be satisfied by a sub-task.

**Procedure:** A unit of behaviour which forms part of a task, or sub-task, and which describes ordered sequences of actions upon objects to reflect different working strategies.

**Action:** The lowest level decomposition of behaviour in the execution of a task, or sub-task, and which is directly carried out on objects and their properties.

**Object:** The lowest level element that can be manipulated within a task, or sub-task, and which is directly operated on by actions.

**Pre-condition:** The necessary conditions under which the task performer can carry out the selected part of the task.

**Post-condition:** The condition that the selected part of a task has achieved once carried out.

---

**Box 20:** The basic concepts used to describe the role-oriented task model (derived from [Johnson et al. 1988, Waddington & Johnson 1989]). Note that the pre-conditions and post-conditions also serve to show those parts of the task that must be carried out sequentially and those parts of the task which can be carried out concurrently and independently. They can also be used to show those parts of the task which are conditional and those parts of the task which are optional.

### 3.3.2. Task Model for Providers

The high-level task decomposition for those agents whose role is to model contribution structures is as follows:

**Role:** A contribution structure provider.

**Task:** To define and maintain the contribution structure underlying the tangible artifacts produced in the RE process.

**Goal:** An up-to-date model of the contribution structure reflecting the social structure underlying the production of requirements artifacts.

**Sub-goals:** Two top-level sub-goals, which we refer to as SG-A1 and SG-A2, whose respective decomposition into procedures can be found in Figures 48 and 49.
SG-A1: Get details about artifacts and agents into system
SG-A1.1: Get details about artifacts and/or artifact-based relations into system
SG-A1.1.1: Get details about artifacts into system
  (Pre: Required artifact does not exist in system but is accessible external to it)
PROC-A1.1.1.1: Import named artifact details
  (Post: Details of artifact available to system)
  (Pre: Required artifact does not exist in system)
PROC-A1.1.1.2: Define new artifact details
  (Post: Details of artifact available to system)
  (Pre: Details of required artifact available to system)
PROC-A1.1.1.3: Select artifact
  (Post: Artifact selected)
  (Pre: Artifact selected)
PROC-A1.1.1.4: Edit artifact details
  (Post: Updated details of artifact available to system)
  (Post: Details about artifacts available to system)
  (Pre: Details about more than one artifact available to system)
SG-A1.1.2: Get details about artifact-based relations into system
PROC-A1.1.2.1: Select artifact(s)
  (Post: Artifact(s) selected)
  (Pre: Artifact(s) selected)
PROC-A1.1.2.2: Define artifact-based relation between artifacts
  (Post: Details of artifact-based relation available to system)
  (Pre: Artifact(s) selected and artifact-based relation available to system)
PROC-A1.1.2.3: Edit artifact-based relation between artifacts
  (Post: Updated details of artifact-based relation available to system)
  (Post: Details about artifact-based relations available to the system)
  (Post: Details about artifacts and/or artifact-based relations available to system)
SG-A1.2: Get details about agents and/or organisational relations into system
SG-A1.2.1: Get details about agents into system
  (Pre: Required agent does not exist in system)
PROC-A1.2.1.1: Define new agent details
  (Post: Details of agent available to system)
  (Pre: Details of required agent available to system)
PROC-A1.2.1.2: Select agent
  (Post: Agent selected)
  (Pre: Agent selected)
PROC-A1.2.1.3: Edit agent details
  (Post: Updated details of agent available to system)
  (Post: Details about agents available to system)
  (Pre: Details about more than one agent available to system)
SG-A1.2.2: Get details about pre-defined organisational relations into system
PROC-A1.2.2.1: Select agent(s)
  (Post: Agent(s) selected)
  (Pre: Agent(s) selected)
PROC-A1.2.2.2: Define organisational relation between agents
  (Post: Details of organisational relation available to system)
  (Pre: Agent(s) selected and organisational relation available to system)
PROC-A1.2.2.3: Edit organisational relation between agents
  (Post: Updated details of organisational relation available to system)
  (Post: Details about organisational relations available to the system)
  (Post: Details about agents and/or organisational relations available to system)
  (Post: Details about artifacts and agents available to system)

Figure 48: The first sub-goal, SG-A1, of the main goal for providers.
(Pre: Details about artifacts and agents available to system)
SG-A2: Get details about contribution relations into system
  PROC-A2.1: Select artifact(s)
    (Post: Artifact(s) selected)
  PROC-A2.2: Select agent(s)
    (Post: Agent(s) selected)
    (Pre: Artifact(s) and agent(s) selected)
  PROC-A2.3: Define contribution relation between artifact(s) and agent(s)
    (Post: Details of contribution relation available to system)
    (Pre: Artifact(s) and agent(s) selected and details of contribution relation
     between them available to system)
  PROC-A2.4: Edit contribution relation between artifact(s) and agent(s)
    (Post: Updated details of contribution relation available to system)
    (Post: Details about contribution relations available to system)

Figure 49: The second sub-goal, SG-A2, of the main goal for providers.

3.3.3. Task Model for End-Users

The high-level task decomposition for those agents whose role is to make end-use of contribution structures is as follows:

Role: A contribution structure end-user.

Task: To uncover various details about those who have contributed to the tangible artifacts produced in the RE process.

Goal: The ability to perform different forms of personnel-based RT.

Sub-goals: Two top-level sub-goals, which we refer to as SG-B1 and SG-B2, whose respective decomposition into procedures can be found in Figures 50 and 51.
(Pre: Details about artifacts, agents and contribution structures available to system)
SG-B1: Formulate trace query
   (Pre: Details about artifacts available to end-user)
SG-B1.1: Compose simple artifact-centred trace query
   PROC-B1.1.1: Select artifact(s)
      (Post: Artifact(s) selected)
      (Pre: Artifact(s) selected)
   PROC-B1.1.2: Request related agents
      (Post: Related agents requested)
      (Pre: Artifact(s) selected)
   PROC-B1.1.3: Request related artifacts
      (Post: Related artifacts requested)
      (Pre: Artifact(s) selected and canned artifact-centred trace queries available to end-user)
   PROC-B1.1.4: Select canned artifact-centred trace query
      (Post: Canned artifact-centred trace query requested)
      (Post: Simple artifact-centred trace query formulated)
      (Pre: Details about agents available to end-user)
SG-B1.2: Compose simple agent-centred trace query
   PROC-B1.2.1: Select agent(s)
      (Post: Agent(s) selected)
      (Pre: Agent(s) selected)
   PROC-B1.2.2: Request related artifacts
      (Post: Related artifacts requested)
      (Pre: Agent(s) selected)
   PROC-B1.2.3: Request related agents
      (Post: Related agents requested)
      (Pre: Agent(s) selected and canned agent-centred trace queries available to end-user)
   PROC-B1.2.4: Select canned agent-centred trace query
      (Post: Canned agent-centred trace query requested)
      (Post: Simple agent-centred trace query formulated)
      (Pre: Details about artifacts and agents available to end-user)
SG-B1.3: Compose more complex trace query
   (Pre: More complex canned trace queries available to end-user)
SG-B1.3.1: Select more complex canned trace query
   (Pre: Required trace query exists in system)
   PROC-B1.3.1.1: Select trace query name
      (Post: Trace query name selected)
      (Pre: Trace query name selected)
   PROC-B1.3.1.2: Provide trace query arguments
      (Post: Trace query arguments provided)
      (Post: More complex canned trace query requested)
      (Pre: Required trace query does not exist in system)
SG-B1.3.2: Define new trace query
   (Pre: Trace query name does not exist in system)
   PROC-B1.3.2.1: Define trace query name and arguments
      (Post: New trace query name and arguments available to system)
      (Pre: Trace query name exists in system but has no defined operation)
   PROC-B1.3.2.2: Define trace query operation
      (Post: New trace query operation available to system)
      (Post: New trace query added to system's canned trace queries)
      (Post: More complex trace query formulated)
      (Post: Trace query formulated)

Figure 50: The first sub-goal, SG-B1, of the main goal for end-users.
3.4. Assumptions and Decisions

Here, we outline some of the assumptions and decisions that were made to satisfy the high-level requirements and operational requirements in the prototype tool:

(1) **Obtain, record, organise, and maintain details about artifacts and artifact-based RT relations.** We assume that all the tangible artifacts produced in the RE process can be input to the system and are maintained within a project repository, as this is increasingly possible with today’s sophisticated document management facilities. We further assume that the project repository deals with the artifact-based RT relations we have suggested, since we have described how tools already exist which are dedicated to this aspect. We only intend to provide limited facilities for the initial construction and editing of artifacts within the system, as this is not our emphasis, and the trend towards more open architectures in which third-party applications can be used for such purposes makes provision unnecessary. Furthermore, the internal structure of these artifacts, along with details about their artifact-based RT relations, will be described within the artifacts themselves using some form of descriptive markup. These markup details can be interactively defined and edited as needed and the markup can be parsed to extract the information necessary to define contribution structures. Nothing further needs to be known about the internal content of these artifacts for our purposes.

(2) **Obtain, record, organise, and maintain details about agents and predefined organisational relations.** We assume that information about the agents involved in the RE process can be input to the system and is maintained within a project repository. Although we are only concerned with recording agent identities, and perhaps some predefined organisational relations between them, we assume that this information can be extended on a project specific basis to include information like contact details, availability, and formal roles.
(3) **Obtain, record, organise, and maintain details about contribution relations.** We intend to append the contribution relations between agents and artifacts interactively once their details are available to the project repository. The granularity of the end points of this relation will be left user-definable. Details of these relations will further be stored with the artifacts concerned, as part of their descriptive markup, and can similarly be parsed to extract the information necessary to define contribution structures. Although we recognise the limitations of defining such relations manually and in a post hoc manner, this procedure is intentionally simplified to demonstrate the basic concepts.

(4) **Model contribution structure.** The markup language in which the artifacts and details of their artifact-based RT relations and contribution relations are to be described will contain all the raw information that is needed to model contribution structures. This information will be extracted from the artifacts which are registered to a project and stored in an underlying factbase of some form. This information will be kept up-to-date as these artifacts or any of their relations are interactively changed. All such changes will be interpreted and immediately reflected in the underlying markup and then in the factbase. Note that we intend to make use of our model-based specification given in Part III of this thesis to define and maintain this model.

(5) **Use contribution structure.** The factbase containing the information about agents, artifacts, and their various interrelations will be able to be interrogated using a simple query language of some form. Note that we intend to make use of some of the query operations given in Part III of this thesis to demonstrate how contribution structures can be used to provide various forms of personnel-based RT. We shall not focus on the visualisation aspects involved in presenting contribution structures, trace requests, and the resulting traces. Canned textual screens will suffice for demonstrating the concepts.

4. Tool Description

In this section, we describe the tool we have prototyped to demonstrate, refine, and evaluate our approach. We first outline its overall architecture and implementation details. We then describe the two main components of the tool in more detail. In so doing, we highlight how they interoperate, and also show how personnel-based RT is made possible through application of the approach. We further convey how the approach could be supported by minimal extensions to those tools already supporting RT or by configuring readily available general-purpose tools. We end with a list of some considerations that would need to be observed in a production version.

4.1. Overall Architecture and Implementation

A schematic of the tool is given in Figure 52. The tool has been implemented on an Apple
Macintosh LC475. It uses a combination of HyperCard and MacPROLOG, as the front-end and back-end respectively. Requirements artifacts are held in an on-line project repository which manages the artifact-based RT relations we identified in Part III of this thesis. As this aspect is not meant to be the focus of the tool, we assume that either some agent is responsible for describing these artifact-based RT relations, or that some of these can be established automatically. These artifacts and their accompanying details are stored in a marked up form in this repository. The project repository also maintains any information provided about the agents participating in a project. As agents and artifacts are registered to a particular project, their associated details become accessible to the main tool, as they get stored in the contribution structure manager's factbase via the parsing that takes place through the traceability extension tool. The traceability extension tool and the contribution structure manager are described in turn below.

![The architecture of the tool.](image)

### 4.2. Traceability Extension Tool

The traceability extension tool is implemented in HyperCard. It provides a hypertextual interface to the information held in the project repository and enables details about artifacts, agents, and their various interrelations to be interactively described for a project. Such details are recorded in the markup of the artifacts concerned. It further provides a way to extract that information required to model contribution structures from these artifacts and to place it in the factbase of the contribution structure manager. In addition, it provides facilities to interrogate the current state of
the contribution structure model, as defined by the information held in the factbase. It therefore acts as the medium through which contribution structures can be defined, maintained, and used.

### 4.2.1. Project and Work Spaces

The traceability extension tool makes use of a virtual project space to delimit those artifacts and agents within the project repository which are registered to specific projects. When a particular project is invoked, the factbase is populated with information specific to that project. Its agents and artifacts are then readily accessible from within the traceability extension tool and can be displayed in the work space of Figure 53. It further becomes possible to navigate between this corpus of project-specific information via hypertext links which follow the project's artifact-based RT relations and contribution relations.

![Figure 53: An empty work space in the traceability extension tool. When an artifact is displayed, its identifier and overall contribution format is displayed. Any internal components and internal contribution formats are also signalled. Any artifact-based RT relations attached to this artifact and its internal components are further signalled and can be traversed if required.](image)

The traceability extension tool also provides the basic text editing facilities needed to create textual artifacts or to edit existing ones. Any new or changed artifacts become registered to the current project, reflected in the project repository, and any details relevant to the contribution structure are further reflected in the factbase. The basic operations available from within the
traceability extension tool for dealing with artifacts, project spaces, and the like, are shown in Figure 54. Those options which have not yet been pointed to will become clearer later on in this section.

<table>
<thead>
<tr>
<th>MCS</th>
</tr>
</thead>
<tbody>
<tr>
<td>About MCS...</td>
</tr>
<tr>
<td>Help...</td>
</tr>
<tr>
<td>Export MCS Data</td>
</tr>
<tr>
<td>New Project...</td>
</tr>
<tr>
<td>Open Project...</td>
</tr>
<tr>
<td>New Artifact...</td>
</tr>
<tr>
<td>Open Artifact...</td>
</tr>
<tr>
<td>Save...</td>
</tr>
<tr>
<td>Save Copy As...</td>
</tr>
<tr>
<td>Close...</td>
</tr>
<tr>
<td>Print...</td>
</tr>
<tr>
<td>Agent Profile...</td>
</tr>
<tr>
<td>Artifact Profile...</td>
</tr>
<tr>
<td>Quit MCS...</td>
</tr>
</tbody>
</table>

**Figure 54:** The basic menu options from the traceability extension tool.

### 4.2.2. RT Markup Language

There is a recent trend within many organisations to apply markup to documents so that they can be stored and retrieved using database techniques. Examples of this trend, along with an explanation as to the added value it brings to documents, are highlighted in [Christophides et al. 1994, IEE 1994]. We assume that the requirements artifacts we are interested in are described in a marked up form, using a descriptive markup language, examples of which are the Standard Generalized Markup Language (SGML) [Goldfarb 1981, ISO 1986] and the Office Document Architecture (ODA) [Horak 1985, ISO 1989]. We suggest descriptive markup because this focuses on the structure of artifacts, as opposed to procedural markup which focuses on their format, since the units to which we wish to append details about related artifacts and contributors are purely structural in nature. This also provides a more desirable starting point from which to build any sort of factbase from the marked up information. As a particular example, the artifact-based RT relations could be described using the Hypertext Markup Language (HTML) instantiation of SGML [Berners-Lee & Connolly 1993], by introducing high-level link semantics like "references", "adopts", and "contains" for the hypertext relations between artifacts. This would mean that the artifact-based RT relations were defined with the artifacts themselves and that the source and target of these relations were hypertextually linked.

However, and for the purposes of the prototype, we make do with a simple RT markup language.
Part IV: Validation

(RTML). This is restricted to focus on defining and editing contribution relations and artifact-based RT relations by direct manipulation of the artifacts in the work space of the traceability extension tool. This markup is not reflected in the presentation of the artifacts concerned, but appended to their underlying descriptive markup using primitive elements like those specified in Figure 55. The traceability extension tool further provides a basic interactive editor and parser for this markup, though this is restricted to parse only that information which is necessary to define and maintain contribution structures. We suggest that RT markup is practical because, if the requirements artifacts were described as HTML documents as mentioned above, the primitive elements of RTML could easily be implemented as extensions to the HTML Document Type Definition.

<table>
<thead>
<tr>
<th>Artifact element</th>
<th>Tags for start and end marker</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artifact name</td>
<td><code>&lt;ART=[Artifact]&gt;</code> <code>&lt;ART=[Artifact]&gt;</code></td>
</tr>
<tr>
<td>Artifact-based relation</td>
<td><code>&lt;AREL=[Source,Target,Relation type,Purpose]&gt;</code> <code>&lt;AREL=[Source,Target,Relation type,Purpose]&gt;</code></td>
</tr>
<tr>
<td>Principal capacity</td>
<td><code>&lt;CP=[Agent,Qualification]&gt;</code> <code>&lt;CP=[Agent,Qualification]&gt;</code></td>
</tr>
<tr>
<td>Author capacity</td>
<td><code>&lt;CA=[Agent,Qualification]&gt;</code> <code>&lt;CA=[Agent,Qualification]&gt;</code></td>
</tr>
<tr>
<td>Documentor capacity</td>
<td><code>&lt;CD=[Agent,Qualification]&gt;</code> <code>&lt;CD=[Agent,Qualification]&gt;</code></td>
</tr>
</tbody>
</table>

**NOTE**

*Rules to interpret and use these elements are defined in the contribution structure manager's inference engine*

**Figure 55:** Some primitives of the RT markup language. Note that both the beginning and end of the structural components are marked using tags enclosed in angular brackets.

The use of markup means that all the information necessary to model contribution structures is stored and maintained with artifacts themselves. Artifacts can subsequently be viewed in the work space, in full or in part, either with or without their underlying markup, or filtered to account for a specified subset of the markup. In this way, all the internal components of a specified artifact that have been contributed by a specified agent could be highlighted. Figures 56 and 57 illustrate the
same artifact without and then with its underlying RTML.

Figure 56: View of an artifact without its RT markup. This is what the users of the traceability extension tool see, edit, and markup.
Note that, to specify the artifact-based RT relations, the RTML delimiters are placed in the marked up master copy of the artifacts concerned at both the beginning and end of the portions related, be they entire composite artifacts or internal components. To specify the contributors to the whole of a composite artifact, the RTML delimiters are placed at the extremes of the artifact concerned. To specify the contributors to the internal components of a composite artifact, the markup delimiters are placed at the beginning and end of the internal component concerned. As such internal components of an artifact become delimited, to account for more granular areas of contribution or to account for end points of artifact-based RT relations, these themselves become tagged with an automatically generated artifact name, until named otherwise, and the necessary containment relations are put in place. These newly formed internal components are henceforth recognised as registered artifacts in their own right by the system. An example of how such an internally delimited component of a composite artifact is depicted in the traceability extension tool is shown in Figure 58. Also note that, when the capacities of the contribution format are defined for an artifact, the default qualifications of pending approval and indifferent are assumed for the principal and documentor capacities, until specified otherwise, whereas the author qualification is determined by taking the nature of any artifact-based RT relations that exist for the artifact concerned into account. For the internal components of a composite artifact, the default incumbents of the three capacities are inferred from the markup of the containing artifact, again until specified otherwise. This is also shown in Figure 58. If any of the above details are interactively changed, these changes are always reflected in the underlying markup of the artifacts concerned.
The basic operations available from within the traceability extension tool that are related to the use of the RTML are shown in Figure 59. Once an artifact or internal component has been selected, the contributors can be assigned from the list of agents registered to the project the artifact is a part of, else new ones can be defined and added to the agent details in the project repository. Their contributions can then be qualified, if desired, noting that the authorial status can be determined automatically. Similarly, any artifact-based RT relations can be put in place, and any further internal components of an artifact can be delimited to do so. From this menu, it is also possible to see the various artifacts currently related to the artifact displayed in the project space, and to further hypertextually jump to them in their own work space as required.
4.3. Contribution Structure Manager

The contribution structure manager is implemented in MacPROLOG. It stores the information described by the RTML of artifacts registered to a project within its factbase. It further contains rules that use this information to infer details regarding default agent capacities, the social roles of agents, their commitments, and so forth. It implements the model-based specification described in Part III of this thesis and is therefore responsible for defining and maintaining a model of the contribution structure. It further implements the operations which provide the ability to query and make use of this model of the contribution structure for personnel-based RT.

4.3.1. Modelling and Exporting Contribution Structures

When a marked up artifact is brought into a project space, when a new artifact is marked up, or when existing markup is changed, the factbase of the contribution structure manager is populated by parsing the artifact's RTML. This extracts those details from its RTML tags which are needed to model the state of the contribution structure and stores them as PROLOG facts in the factbase. The artifact identifier and any component artifacts are extracted to define the containment relations. The artifact-based RT relations are extracted to define the connectivity relations. Contribution formats are extracted to define the contribution relations. Rules to infer the default incumbents of capacities, the authorial status, and so forth, are automatically fired as required. Once any of the RTML tags are subsequently changed, or new tags are appended, the state of the contribution structure is redefined according to those operations we have specified which change its state. These operations are stored as PROLOG rules in the inference engine of the contribution structure.
structure manager. In this way, the RTML can be seen to provide a dynamic link between information in the project repository and the factbase.

To enable the further forms of analysis that we mentioned in Part III of this thesis, the system is also able to export the fundamental information needed to define the contribution structure as comma delimited files. In such a format, this information can thereby be input to those third-party databases which support social network analyses and the like. It is worth pointing out here that, although the traceability extension tool could directly interface with existing relational databases like Oracle, MacPROLOG was chosen for convenience.

4.3.2. Using Contribution Structures

Those operations we specified that act upon the state of the contribution structure model to perform various forms of personnel-based RT query are also stored as PROLOG rules in the inference engine of the contribution structure manager. Personnel-based RT queries can be formulated within the traceability extension tool and then sent to the contribution structure manager. This then makes use of the rules for the query-based operations and the current state of the contribution structure model to generate a response. The response is then sent back to the traceability extension tool where it is displayed. Alternatively, by selecting and querying an artifact in the traceability extension tool, its artifact profile can be determined and displayed. This retrieves relevant information from the contribution structure manager to provide details about all the artifacts and agents related to the artifact in question, as shown in Figure 60. Agent profiles can be provided in a similar way, as shown in Figure 61. As each element in these profiles are themselves hypertext anchors, artifact and agent profiles act as one of the many navigational springboards from which to undertake various forms of RT.
Part IV: Validation

Figure 60: The details provided in an artifact profile.

Figure 61: The details provided in an agent profile.
In Figure 62, we indicate the queries we have implemented within the tool. These are some of the artifact-centred, agent-centred, and agent/artifact-centred query-based operations described in Part III of this thesis. They enable the contribution structure underlying a project to be viewed in multiple ways and in selective detail. For instance, it is possible to view: (a) all of a specified agent's contributions; (b) all of a specified agent's contributions when acting in a specific capacity; (c) all of a specified agent's contributions with respect to the history of a specific artifact; (d) all relayed artifacts on behalf of a specified agent; (e) all approved artifacts and their principals; and so forth.

**Figure 62:** Query-related menu options from the traceability extension tool. Such traces are made possible through its interface to the contribution structure manager, where rules for these queries are defined and can be invoked.

In Figures 63 through to 65, we go on to show some of the ways in which the results to the above type of query given in Figure 62 are presented in the traceability extension tool once answered. Note that the results are presented crudely and textually, since trace visualisation is not the focus of this thesis. Also note that instantiation of these templates with project information is only given when we describe a scenario in the following section.
Figure 63: Reporting on a forwards RT request. The information provided as a result of interrogating the contribution structure manager from within the traceability extension tool is further displayed in the traceability extension tool.

Figure 64: Reporting on the various relations between specified artifacts. The information provided as a result of interrogating the contribution structure manager from within the traceability extension tool is further displayed in the traceability extension tool.
Figure 65: Reporting on the various relations between specified agents. The information provided as a result of interrogating the contribution structure manager from within the traceability extension tool is further displayed in the traceability extension tool.

4.4. Considerations for a Production Version

What we have described above is a prototype. To develop a production version, there are many further considerations that would need to be taken into account, like:

(1) To support heterogeneous requirements artifacts, and their subsequent change, it would be preferable to make use of an open multimedia environment in which an artifact's host application can take over control for its editing and maintenance.

(2) To deal with multiple artifacts and their interrelations concurrently, it would be preferable to make use of a multi-tasking windowed environment.

(3) RE is a social process that can involve much, possibly distributed, group working. To enable multiple agents to be simultaneously involved in defining, maintaining, and using contribution structures, it would be preferable to provide more multi-user facilities.

(4) Each project is likely to need a customised approach. It would be preferable to provide more facilities to customise the approach on a project-specific basis, say to account for any project-specific artifact-based RT relations, personnel-based RT required, and so forth.
Even without constructing a special-purpose tool, we believe the approach could readily be made operational by configuring general-purpose tools to support it. Our prototype itself demonstrates how it could be supported by extensions to standard document preparation systems and database systems. Moreover, where tools to support RT are currently used within an organisation, the approach could probably be supported by minimal extensions to the RT schemes they are intended to support.

5. Scenario of Use

In this section, we provide a scenario to illustrate how the prototype supports the approach. In particular, we illustrate how use of the approach makes it possible to uncover various details about the social dimension of RE that would otherwise remain obscured, and we further indicate how the availability of such details can inform practice.

5.1. Description of Scenario

The text for the scenario is given in Box 21. In the subsequent discussion, we only intend to illustrate some of the information that can be obtained from modelling the contribution structure underlying the RE exercise described in this scenario. Specifically, we intend to show how the approach provides a way for member 3 of the group of requirements engineers to get an overall picture in which to understand the change that occurred between the two versions of the requirements specification, to be able to locate those involved, and so be in a position to address the problem encountered in conjunction with the most suitable agents. In this way, we highlight how conventional notions of artifact-based RT can be extended with associated contribution structures to enable personnel-based RT.
A software project began with a wish list, reporting the needs from a group of users, which was written up by a scribe and authorised by a project leader. The project leader then held a meeting, of which an audio tape record was made, to discuss the wish list with a group of stakeholders. A direct transcript of the meeting was subsequently made by a couple of secretaries. From the transcript and the wish list, along with numerous other input documents, an initial requirements specification was written by a group of requirements engineers being managed by the project leader. Each requirements engineer concentrated on different parts of this document. Member 1 composed paragraph x.

Following circulation to, and comments from, various interested parties, a revised version of the requirements specification was written. In particular, an alteration had been made to paragraph x as a result of an email message from the managing director's personal assistant to the project leader. In this message, the managing director passed on a verbal change request she received from user 1, a member of the original group of users. This corrected version of paragraph x becomes paragraph y in the revised requirements specification.

Unfortunately, member 2 of the group of requirements engineers inadvertently introduced an error when carrying out this change, largely because he did not acknowledge the subtlety of the wording in the particular fragment of the email message detailing the change request. This was because he had not been involved in the original discussion about the requirement at issue and had assumed that the managing director was being unnecessarily fussy with wording. In checking the revised requirements specification, member 3 of the group of requirements engineers noticed the problem with the requirement specified in paragraph y.

**Box 21: Scenario text. Note that the agents are in bold font and the artifacts are underlined.**

In following the discussion of the scenario, it may be helpful to refer to Figure 66 which outlines the steps of the approach, repeated from Figure 16 of Part III. In this discussion, we describe how each of the steps can be applied to the scenario, and illustrate the information that is captured in so doing.
5.2. How the Approach Represents the Scenario

Firstly, the artifact-based RT relations are defined. Figures 67 and 68 show the basic temporal and developmental relations that can be said to exist between the artifacts produced in the scenario.
Figure 67: The historical order of artifact production in the scenario.
The concern of most conventional artifact-based RT schemes lies within this box. The requested change might be integrated if documented in a project change request form and could be accessible between the two versions of the RS.

Figure 68: The developmental order of artifact production in the scenario.

The schematic of Figure 69 shows how the information about the scenario can be represented by following the approach. It shows some of the additional containment and connectivity relations that can be said to exist between the artifacts. It also gives minimal semantics for these relations. For example, as paragraph y corrects paragraph x, it is altering pre-existing content, so the semantics of this connectivity relation is that of adopts. Note that the above temporal and developmental relations could also be represented by giving appropriate relations and semantics here. The prototype tool can handle these different relations and their semantics and further makes some deduction possible. For instance, on querying the trace behind paragraph y, these relations can be used to locate the email message as the reason for the change from paragraph x, whilst also retrieving the various derivation paths from its origin as the requirement in the wish list. Such a rule to find artifacts causing change is shown in Figure 70.
Figure 69: The auxiliary relations and contribution formats in the scenario. This figure is meant to be suggestive of the kinds of auxiliary relations that could be said to exist between the artifacts produced in the scenario and by no means intends to exhaust all the possibilities. These relations can be used to provide an increasingly granular picture as to how the artifacts produced in the scenario fit together and can therefore be chosen dependent on specific project needs.
Part IV: Validation

Secondly, Figure 69 illustrates how the contribution format can be defined for the artifacts produced in the scenario. The single underlining signifies those capacity incumbents which can be automatically determined from the containment relations and the double underlining signifies those capacity incumbents which can be automatically determined through versioning. Only those agents that are not underlined are those that need to be added manually at present. However, we anticipate that it is also possible to automatically determine some of this information too. With this added information, this means that each of the relevant artifacts retrieved as a result of the trace query we just mentioned on paragraph y can be augmented with their associated contributors, further indicating the capacities in which they have contributed, as well as those other agents with whom they have ties. This information points out aspects like: (a) when member 2 first became involved with the project in general and with the requirement documented in paragraph y in particular, as an individual and as a group member, and in what capacities; (b) which agent was involved in the same social role as member 2 with the previous version of the requirement documented in paragraph y, namely member 1; and (c) member 2's relations to member 1, like those defined through their areas of direct collaboration.

For clarity, we have not qualified the capacities of the contribution format in Figure 69. However, we can see that, although member 2 is the author of paragraph y, he is in fact altering member 1’s authored contribution of paragraph x because of the MD’s authored contribution in the email.
message. We can thereby extract the authorial dependencies between these agents. We could also examine the changing degrees of commitment to what is documented as the requirement if the mood of the documentor had been given, say if a mood shift had occurred between paragraph x and paragraph y from indifferent to uninformed. Furthermore, if paragraph y were pending approval by the principal, this would mean that the change has yet to be signed off, and so signal no forward repercussions of the change if it is policy in the project for artifacts to be approved before being used elsewhere. A record of those artifacts that have been approved and not approved, and who by, would itself be a useful source of information for later reference.

By manipulation of the above details, the prototype tool can determine information about the social roles and role relations of all those involved, from which it can infer further details about individual, collective, and social commitments. From this extra information, the same query of paragraph y can reveal various information related to the social structure underlying its production process. For instance, we are alerted to the fact that: (a) the MD was acting on behalf of user 1 when requesting the change, as their respective social roles of ghost author and sponsor of the particular email fragment documenting the change request delineate where their respective commitments lie, so she is only superficially the change instigator; (b) the basis for this role relation between the MD and user 1 can be found in their joint collaboration in the original project meeting, to which member 2 was not party, so member 2 is unlikely to be aware as to the full extent of their joint involvement with the particular requirement which they have requested changes to; and (c) user 1 is the original source of the content and motive for the requirement documented in paragraph y, as the devisor of the requirement from which it has been derived in the wish list, so is the agent who is ultimately committed to its realisation and effect, thus giving user 1 the authority to request the change.

5.3. What the Approach Brings to the Scenario

We repeat some of the salient points in the above discussion here, along with excerpts from within the tool, to illustrate the ways in which this extra information becomes available. When a trace is retrieved from paragraph y back to its original source artifact, the various paths obtained are augmented with details of their contributors at each stage, as shown in Figure 71. So, where the problem is a consequence of member 2 changing the content of paragraph x in the initial RS to form paragraph y in the revised RS, this enables us to see where member 2 first became involved in the work resulting in paragraph y. It also enables us to identify in what capacities member 2 contributed, namely as author and documentor, and to identify the previous artifact in the developmental chain, namely paragraph x which paragraph y alters. It is thereby possible to find out which agent was involved in the same social roles as member 2 at the previous stage of the requirements life, namely member 1. It is also possible to uncover the trail of artifacts which are referenced as also having something to do with the transition between paragraph x and paragraph y.
Part IV: Validation

and to further be able to identify the various agents involved with these.

Figure 71: Trace to the original artifact source of paragraph y in the scenario, with the associated contribution formats retrieved along the way. Note that it would be preferable to display such information in a graphical manner for clarity, though trace visualisation is not a concern of this thesis.

Figure 72 shows the artifact profiles that can be revealed for those artifacts we are most concerned with in the scenario. Such profiles collect together information anchored about a specific artifact. They show the agents who contribute to the artifact, how they contribute, and the qualification of their contributions. They also show the social roles amongst the contributors, which delineates their commitments, as well as the various dependencies between the artifact profiled and other artifacts. Here, these profiles can be used to reveal RE member 2's relationship to, and previous involvement with, member 1. Member 2 is the author of paragraph y, but has authored by altering paragraph x, which in turn was authored by member 1. Dependencies like this can be traced to any depth required if we recall that all the elements in such profiles are hypertext anchors from which personnel-based RT queries can be pursued.
Similarly, Figure 73 shows the profiles of those agents we are most concerned with in the scenario. Such profiles collect together information anchored about a specific agent. They identify aspects such as an agent's involvement with artifacts and in which capacities, with agents and in which role relations, and so forth. They therefore provide details of social roles, collaborators, and other related agents. Here, these profiles can be used to reveal RE member 2's individual contributions and joint contributions as a group member, so the agents he has collaborated with, on what, and under what arrangements. Again, the hypertextual interface means that the relations formed between specific agents throughout a project can be pursued.
Furthermore, when paragraph y is directly queried, information like that shown in Figure 74 can be obtained. This provides information relating to any changes that have occurred, such as: (a) what and who instigated the change; (b) who requested it; and (c) who authorised it. Here, we are alerted to the fact that paragraph y alters paragraph x due to a particular fragment in the email message. Also, that the MD is only the superficial change instigator, as she issued the change request on behalf of user 1.
As previously mentioned, each of the elements in Figure 74 is a hypertext anchor which can allow these change details to be pursued further. These can therefore be used to trace more details about the various relations between the specific agents associated with the change, as shown in Figure 75. These can also be used to trace more details about those artifacts instigating the change, as shown in Figure 76. Such an ability to rapidly identify appropriate agents and artifacts can help eliminate many of the issues associated with the requirements change process, like identifying where the responsibility for change lies, speeding up the authorisation of change, and communicating change to all the relevant parties [Chudge & Fulton 1994].
Figure 75: Following up change details by looking at the relations between the agents concerned in the scenario, namely between the M.D. and user 1.

Figure 76: Following up change details by looking at the artifact that instigated the change in the scenario, namely the fragment in the email message.
The above discussion serves to illustrate only some of the sorts of personnel-based RT query that can be carried out as a consequence of modelling the contribution structure underlying the RE process described in the scenario. Moreover, it serves to indicate how these extensions can complement conventional forms of artifact-based RT. It shows how, from the provision of rudimentary input data, the approach can be used to construct a rich picture of a potentially complex social structure and make explicit information that would otherwise remain hidden or be incorrectly inferred.

6. Case Study

In this section, we present a case study based on a real, albeit small-scale, industrial RE project. Its objective is to attempt to evaluate the approach using real project data in conjunction with real practitioners. We first outline some details about the case study and then show how the contribution structure underlying the project was modelled and used by following the approach. We show how personnel-based RT is made possible, which enables us to answer the frequently asked questions given in Part III of this thesis, and we further indicate how it offers the potential for many interesting forms of analysis. We then point to some of the issues which arose when conducting the case study and we finally discuss the benefits and drawbacks of the approach as seen from a practitioner's point of view. Throughout this section, the name of the company, its employees, its customers, and its suppliers have been changed to maintain confidentiality.

6.1. Description of the Case Study

We first describe the company and the project from which we gathered data. We then describe the data that we gathered about the project, along with how this was captured. We finally list details about the agents and artifacts which made up the project.

6.1.1. Company and Project

The project that is the subject of our case study comes from a small communications company which employees about twenty-five people. Henceforth, we will refer to this company as "the Company" and to the project as the "DR project". The Company concurrently run many projects and are in the business of providing both software and procedural solutions to many communications-related problems. The DR project was initiated in February 1992 and continues to evolve to date.

Back in February 1992, the original objective of the DR project was to supply an existing customer, a large banking institution, with a dedicated communications service to complement their existing DR program. This service was intended to deal with all the bank's in-coming and out-going communications, as well as to inform its customers and staff of the state-of-play in any
situation under which the bank's computing services encountered problems, or if the physical buildings it occupies had to be evacuated. This service went live at the end of March 1992 and continues in exactly the same form, for this specific customer, over three years later.

Around the middle of August 1992, the idea of developing a more generic DR service to offer to other potential customers was discussed. Six versions of a requirements and design specification were consecutively drawn up for this service throughout September 1992. Work on developing this generic service then subsided. It picked up again at the end of October 1992. This was because specific individuals had now been employed by the Company to finalise the details of this service, based on the requirements and design specification, and to also market the service. Following a fair amount of staff turn-over, the first new customer did not sign-up to this generic DR service until February 1994. Throughout the period from October 1992 to February 1994, the requirements and design specification evolved into an operations manual describing the method to be followed to provide the service, accompanied by a manager's guide providing high-level instructions about the service.

Since February 1994, many more customers have subscribed to this generic DR service. As a direct consequence, the service itself has undergone continuous modification to address the additional requirements for new customers, something which has been reflected in both the operations manual and the manager's guide.

6.1.2. Data Gathered

Some of the work that occurred on the DR project, from the initial discussion about providing a generic DR service through to the sixth version of the requirements and design specification, was followed closely. This was the period from the middle of August 1992 to the end of September 1992. We observed all the meetings that took place, as well as took notes and made audio recordings, and collected photocopies of all of the tangible artifacts that were produced in this period. We even actively participated in a few aspects of the process.

During this time, a detailed picture of what had happened back in February and March 1992 when developing the initial customer-specific service, was slowly reconstructed with those who had been involved. However, most of the actual artifacts that had been produced throughout this earlier period had either not been kept or had been filed in unknown locations, exceptions being the final document outlining the specific DR service, the contract with the customer requesting it, and some relevant letters and facsimiles.

After the sixth version of the requirements and design specification was produced at the end of September 1992, the project ground to a halt for a while. From the end of October 1992 onwards, the work was left more or less in the hands of specific individuals affiliated to the sales department.
of the Company, and fewer tangible artifacts were produced. This was because the individual's job involved developing and selling the generic DR service, along with other sales commitments. Throughout this period, we attempted to maintain some record of the main artifacts that were produced as a consequence of the earlier requirements and design specification, though many of the primary artifacts were under construction for months. The peripheral artifacts, like the many sales letters and responses, notes from the meetings with potential customers, and the often endless correspondence when negotiating contracts, are not fully represented in our account. This is because these have been difficult for an outsider to keep track of over the years. Therefore, we are only in a position to estimate some of what truly happened from the end of October 1992 onwards.

6.1.3. Agents

Fifty eight agents directly contributed to the DR project. We refer to these agents using the alphabetic identifiers of AA through to CF in this section. In Appendix D.1., we list the twenty seven agents who participated as the Company's employees, list their formal positions within the Company, and give their dates of employment. We also list the six group labels which were significant to the Company for the DR project, along with their members. It must be noted that not all of the Company's employees contributed to the DR project, notably absent being those from its technical division. Finally in Appendix D.1., we list the twenty five remaining agents who directly contributed to the DR project from outside the Company, including customers and suppliers amongst others. Note here that, as it was often more difficult to obtain the contact details of specific individuals from within these external companies, we often just refer to the company as a whole.

6.1.4. Artifacts

One hundred and sixty six tangible artifacts were produced in the DR project, remembering that we have excluded a large number of the memos and correspondence with customers and suppliers in the later phases of the project. We refer to these artifacts using the numeric identifiers of 1 through to 166 in this section. We have split these artifacts into those produced in four distinct project phases for clarity:

- **Phase one** Those artifacts that were produced from the end of February 1992 through to the end of March 1992. These are to do with providing the initial customer-specific DR service. Twenty three artifacts were produced in this phase.

- **Phase two** Those artifacts that were produced from the middle of August 1992 through to the end of September 1992. These are to do with developing a baseline specification, sufficient to outline the service, and from which others could work. Note that we have the exact dates for the artifacts produced during this period since we were
able to observe the process first-hand. Sixty five artifacts were produced in this phase.

**Phase three** Those artifacts that were produced from the end of October 1992 to the end of July 1993. These are to do with developing a generic DR service which meets the initial requirements and is marketable. Note that we later split the artifacts produced in this phase into two parts to coincide with the change in the project leader. Thirty nine artifacts were produced in this phase.

**Phase four** Those artifacts that were produced from the beginning of September 1993 to the end of June 1995. These are to do with selling and extending the scope of the generic DR service to address new customer requirements. Note that again we split this phase into two parts to coincide with the change in the project leader. Thirty nine artifacts were produced in this phase.

A description of all these artifacts can be found in Appendix D.2.

**6.2. Application of the Approach**

Throughout the course of the DR project, most of the artifacts that were produced were informal and paper-based, and few were retained by those involved. All that remains from this project within the Company today is an early requirements and design specification, an up-to-date operations manual, an up-to-date manager's guide, a number of contracts with customers, and miscellaneous correspondence. Consequently, RT has not been implemented and maintained in the DR project, and we have found that those who had been involved throughout no longer remembered from where or from whom the various aspects of the final DR service had been derived. Some problems have resulted from this loss of information but, because the project is relatively small in size and scope, because the team is small and some agents have been continuously involved, and because the final service does not include any software components, these have not been critical to its maintainability.

Here, we outline how the approach can be applied in the DR project. This has been applied in a post-hoc manner since the approach was not formed when the project started back in 1992. The following details have therefore been constructed in conjunction with some of the key participants, using notes taken throughout the course of the project, and using miscellaneous project notes and dates that had been kept within the Company itself. We first put in place the basic artifact-based RT relations which could be said to exist between the artifacts produced in each of the four phases. We also provide these relations with minimal semantics. We then append the contribution format to each of the artifacts. From the provision of just these details, we point to the kinds of information that can now be inferred about the role relations and commitments of those who had
been involved in the DR project.

6.2.1. Artifact-Based RT Relations

For each phase of the DR project, we examined the order in which the artifacts were produced, and further clarified the temporal relations between them. Note that some guesswork had to be applied in phases three and four because of their long duration. We also determined the coarse flow-down of information amongst these artifacts and indicated how certain artifacts influenced others. We have not determined all the possible relations that could be said to exist between these artifacts, like the transitive relations and the relations between internal components, and have only included the obvious relations that key participants in the DR project were able to identify or confirm. A key participant was further charged to determine whether the communicative function of each relation was one of adopting or referencing, and then encouraged to describe its communicative purpose in more detail.

The artifact-based RT relations that were established for phases one through to four are shown in Figures 77 through to 80. Note that the artifacts are numerically labelled according to the chronological order in which they were produced. Where such information was obscured, we have estimated when artifacts became available to the project. Also note that these figures do not attempt to represent for how long each artifact was in production.
Figure 77: The order in which artifacts 1 to 23 were produced and the predominant artifact-based RT relations between them.
**Figure 78:** The order in which artifacts 24 to 88 were produced and the predominant artifact-based RT relations between them.
Figure 79: The order in which artifacts 89 to 127 were produced and the predominant artifact-based RT relations between them. Note that artifacts 89, 90, 101, and 102 are elongated because they have been used to gather together all the correspondence between the Company and potential customers, since such artifacts were continuously in production and we were unable to keep track of them all. Also note that some other artifacts are slightly enlarged. This signifies that other artifacts were produced whilst they were in production.
Figure 80: The order in which artifacts 128 to 166 were produced and the predominant artifact-based RT relations between them.
In the following description about the artifact-based RT relations, we only focus on phase one in detail. Phases two, three, and four are covered in the same detail in Appendix D.3.

To assist in the comprehension of Figure 77 above, we clarify the temporal relations between the artifacts produced, as shown in Figure 81. We then clarify the nature of the artifact-based RT relations given in Figure 77, as determined by a key participant, in Table 17. In the first column of this table, we list the communicative function between the two related artifacts. In the second column, we list the finer communicative purpose of this relation. In the third column, we list the more specific reason given for the relation. Note that the key participant selected the nature of these relations in accordance to our classification for artifact-based RT relations.

<table>
<thead>
<tr>
<th>Temporal relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before(1,2) Before(6,8) Meets(13,14) During(20,19)</td>
</tr>
<tr>
<td>Before(2,3) Before(8,9) Before(14,15) Before(19,21)</td>
</tr>
<tr>
<td>During(4,3) Before(9,10) Before(15,16) Before(21,22)</td>
</tr>
<tr>
<td>Before(3,5) Before(10,11) Meets(16,17) Before(22,23)</td>
</tr>
<tr>
<td>Before(5,6) Equal(11,12) Before(17,18)</td>
</tr>
<tr>
<td>During(7,6) Before(12,13) Before(18,19)</td>
</tr>
</tbody>
</table>

**Figure 81: The temporal relations between artifacts 1 to 23.**

<table>
<thead>
<tr>
<th>Communicative function</th>
<th>Communicative purpose</th>
<th>Informal description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 adopts 1</td>
<td>2 adds to 1</td>
<td>2 qualifies 1</td>
</tr>
<tr>
<td>3 references 2</td>
<td>2 frames 3</td>
<td>2 is the reason for 3</td>
</tr>
<tr>
<td>4 adopts 2</td>
<td>4 adds to 2</td>
<td>4 defines 2</td>
</tr>
<tr>
<td>6 references 2</td>
<td>2 frames 6</td>
<td>2 is the reason for 6</td>
</tr>
<tr>
<td>4 references 3</td>
<td>3 substantiates 4</td>
<td>3 assists with 4</td>
</tr>
<tr>
<td>5 references 4</td>
<td>5 matched with 4</td>
<td>5 is compared with 4</td>
</tr>
<tr>
<td>6 adopts 4</td>
<td>6 alters 4</td>
<td>6 refines 4</td>
</tr>
<tr>
<td>6 references 5</td>
<td>5 substantiates 6</td>
<td>5 assists with 6</td>
</tr>
<tr>
<td>4 references 7</td>
<td>4 causes 7</td>
<td>7 responds to 4</td>
</tr>
<tr>
<td>8 references 6</td>
<td>6 frames 8</td>
<td>6 is the reason for 8</td>
</tr>
<tr>
<td>8 references 7</td>
<td>7 frames 8</td>
<td>7 is background for 8</td>
</tr>
<tr>
<td>9 references 8</td>
<td>8 causes 9</td>
<td>9 is a result of 8</td>
</tr>
<tr>
<td>10 references 9</td>
<td>9 substantiates 10</td>
<td>9 assists with 10</td>
</tr>
</tbody>
</table>
Based on Table 17, we duplicate Figure 77 to clarify which of its relations are likely to be maintained by contemporary approaches to RT. As these generally capture the developmental flow-down between requirements, as well as their historical order, they are only likely to account for the adopts relations of Table 17, noting often without giving them with any further semantics. The outcome is shown in Figure 82. We then duplicate Figure 77 again in Figure 83 to clarify those additional relations that our approach brings into the picture, namely the references relations.
of Table 17. This is given to indicate the wealth of information that is generally missed with conventional approaches to RT. This is also given to show how contextual information for requirements flow-down, such as those artifacts providing rationale or background, is critical information that needs to be taken into account in RT schemes.
Figure 82: Those artifact-based RT relations that are likely to be maintained between artifacts 1 to 23 by conventional RT schemes, though without the same semantics. The arrows here suggest the flow-down of artifact content.
End of February 1992

End of March 1992

Time

1

2 frames 3

3 frames 4

5 substantiates 6

6 frames 8

8 causes 9

9 substantiates 10

10 frames 11

10 causes 13

11 frames 15

15 frames 16

16 substantiates 18

18 frames 19

19 substantiates 21

21 causes 22

22 causes 23
**Figure 83:** Those further artifact-based RT relations that can be brought into the picture between artifacts 1 to 23. The arrows here suggest the direction of influence between artifacts.

### 6.2.2. Contribution Format

For each phase of the DR project, we identified the principal(s), the author(s), and the documentor(s) of the artifacts produced. Throughout the course of the DR project, we recorded those agents who had been involved in the production of artifacts where possible, and also made a record of to whom these artifacts were distributed. The capacities in which those involved had contributed was then later determined in conjunction with some of the key participants. Note that, because we do not go into detail about the internal components of artifacts and their many interrelations in this case study, we do not pin-point the more specific contributors to internal components here. In Table 18, we list the contributors to each of the artifacts produced in phase one and present an alternative picture of this information in Table 19. Phases two, three, and four are covered in the same detail in Appendix D.4.
Table 18: The contribution formats for artifacts 1 to 23.
Table 19: Contributors to artifacts 1 to 23. In this table, P, A and D depict contributions as principal, author, and documentor respectively. Where these are given in lower case, this means that the contribution was as a group member, with the group identifier also being given in upper case.

6.2.3. Qualification of Contribution Format

The three capacities of the contribution format have not been qualified further in this case study. This is because, if we were to qualify them according to the attributes we suggested, we should have collected the appropriate information from the start of the project. We could qualify the principal contributions retrospectively, to reflect whether artifacts have been approved, abandoned,
or are pending approval, and so would be able to pull out trails of those artifacts pending approval by certain agents, identify those involved with the most abandoned artifacts, and so forth. However, it would be a difficult exercise to retrospectively obtain details about the documentor's mood of transmission for each artifact. Therefore, we only point to how the status in which the authors have contributed can be qualified here, and then show the uses to which such extra information can be put.

With knowledge of the artifact-based RT relations between the artifacts, we can see that out of all of the project artifacts, only artifacts 1, 99, and 100 do not adopt or reference other artifacts. Their respective authors, agents BI, AA/AE/AD/AG, and BB/AP/AQ, can therefore be qualified as "creators". Note that we use the form AA/AE here to account for jointly contributing authors on an artifact. We can similarly identify those authors qualified as adopters, referencers, or both. With such information, we can see which agents produce the most original artifacts, which use their own previous artifacts most often, and which use another agent's contributions most often. For example, agent AH was the project leader during the production of artifacts 89 to 98 in phase three, and was the author of artifacts 89, 90, 91, 92, 93, 94, 95, 97, and 98. Of his authored contributions, he was an adopter in artifacts 93 and 94, a referencer in artifacts 89, 90, and 91, and both a referencer and adopter in artifacts 92, 95, 97, and 98. In artifact 98 he adopted agents AA/AE/AT/AU/AP's authored contribution and in artifact 91 he referenced both agent AG's and agents AE/AA/AC/AG's authored contributions. We can see that none of his authored contributions were adopted by other authors after he left the DR project and only once were any of them adopted by others whilst he was still active in the project. We can also see that artifacts 89, 97 and 98 were each referenced only once, and by artifact 101 after agent AH left the project. Furthermore, this was by author AI who took over agent AH's position as project leader. So, in using these different types of qualified authorial trail, we can begin to assess the influence of an agent's authored contributions on the surrounding body of artifacts, and also on the larger project.

To give an example of a full authorial trail we consider artifact 106 authored by agent AI. This artifact was the first version of the operations manual for the generic DR service. From artifact 106, it is possible to use the artifact-based RT relations to trace all the paths back to the original author(s) and to locate the author(s) of all the ultimate artifacts resulting from it. The authors involved along the way, including information about how each progressive author made use of the previous author's contribution, can thereby be uncovered. In Figure 84, we illustrate part of this authorial trail, but filter it to highlight only those authors whose contributions were adopted in getting to artifact 106. In Figure 86, we illustrate more of this authorial trail, but this time filter it to highlight only those authors who have adopted artifact 106 in their subsequent contributions.
Figure 84: Authorial trail to artifact 106, after backwards RT via adopts relations.
Part IV: Validation

Figure 85: Continuation of an internal authorial trail in Figure 84.

Figure 86: Authorial trail from artifact 106, after forwards RT via adopts relations.
6.2.4. Social Roles and Role Relations

The social roles that agents have assumed when contributing to the artifacts produced in the DR project can be inferred from the information we have so far gathered. Details of the consequent role relations between agents when they jointly contribute to artifacts sheds more light on the contribution structure underlying the project. Not only can we see who has collaborated with whom, but we can see how they have collaborated, and whether or not the role relations between agents have been sustained throughout their joint efforts.

To explain the use of such inferred information, we compare the social roles of two of the project leaders whilst contributing to the DR project, agents AI and AJ. Agent AI was the project leader when artifacts 99 to 127 were produced and was a contributor to twenty two of these. Agent AJ was the project leader when artifacts 128 to 162 were produced and was a contributor to twenty six of these. Their social roles when contributing to these artifacts and their role relations to other agents are shown in Tables 20 and 21 respectively.

<table>
<thead>
<tr>
<th>Social role</th>
<th>On how many artifacts</th>
<th>How many of these were on own</th>
<th>Social roles of other contributors collaborating with agent AI</th>
</tr>
</thead>
</table>
| True author        | 16                   | 13                           | True author = BM  
Ghost author = AA/AE/AD  
Ghost author = AA/AE/AD/AT |
| Nominal author     | 2                    | 0                            | Ghost author = AA/AE/AD/AG  
Ghost author = BB/AP/AQ |
| Representative     | 2                    | 0                            | Sponsor = AD (x2) |
| Ghost author       | 2                    | 0                            | Sponsor = AD and Relayer = BO (x2) |
| Devisor            | 0                    | 0                            | None |
| Sponsor            | 0                    | 0                            | None |
| Relayer            | 0                    | 0                            | None |

*Table 20: The social roles and role relations for agent AI.*
From these tables, we can see that agent AI worked on his own on over half of the artifacts he contributed to, else he worked with small groups of agents. As he worked largely as a true author, he appears to have been a self-sufficient documentor. It is interesting to notice that agent AD only collaborated with agent AI as a sponsor, and only when dealing with the customer-related artifacts. In contrast, we can see that agent AJ rarely worked on his own, and mainly collaborated with one or two other agents. He had a strong dependency on agent AE as his ghost author and on many others as relayers, the latter fact possibly hinting at the need for more sales support. It is interesting to note that agent AD was ultimately responsible for about a third of the artifacts that agent AJ had contributed to.

There could be many reasons to account for the subtle differences in how these two agents with the same institutional role worked. Agent AI did not close any sales and focused on developing a marketable service. In contrast, agent AJ focused on selling the service that agent AI had developed, and only made a few subsequent additions to account for new customer requirements. Notably, it was with such additions that agent AE collaborated with agent AJ as ghost author, perhaps to maintain some continuity as agent AE had also occasionally collaborated with agent AI as ghost author.
6.2.5. Agent Commitments to Artifacts and to Each Other

For completeness, we summarise the kind of information that can be inferred about the commitments of those agents who have contributed in the DR project. For example, in Table 22 we can see which artifacts agent AP is committed to, as well as for which of their aspects. We can immediately see that she is mainly responsible for the physical appearance of artifacts, never for their content on her own, and never for their ultimate effect.

<table>
<thead>
<tr>
<th>Artifact committed to</th>
<th>Aspect of artifact committed to</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Content (as one of many contributors)</td>
</tr>
<tr>
<td>22</td>
<td>Physical appearance (on own)</td>
</tr>
<tr>
<td>31</td>
<td>Physical appearance (on own)</td>
</tr>
<tr>
<td>41</td>
<td>Content (as one of two contributors)</td>
</tr>
<tr>
<td></td>
<td>Physical appearance (on own)</td>
</tr>
<tr>
<td>85</td>
<td>Content (as one of two contributors)</td>
</tr>
<tr>
<td>96</td>
<td>Content (as one of many contributors)</td>
</tr>
<tr>
<td></td>
<td>Physical appearance (as one of two contributors)</td>
</tr>
<tr>
<td>100</td>
<td>Content (as one of many contributors)</td>
</tr>
<tr>
<td>111</td>
<td>Content (as one of many contributors)</td>
</tr>
<tr>
<td>139</td>
<td>Physical appearance (as one of two contributors)</td>
</tr>
<tr>
<td>140</td>
<td>Physical appearance (on own)</td>
</tr>
<tr>
<td>148</td>
<td>Physical appearance (on own)</td>
</tr>
<tr>
<td>151</td>
<td>Physical appearance (on own)</td>
</tr>
<tr>
<td>154</td>
<td>Physical appearance (on own)</td>
</tr>
<tr>
<td>155</td>
<td>Physical appearance (on own)</td>
</tr>
</tbody>
</table>

Table 22: Agent AP's artifact commitment store.

In Table 23, we can further see which other agents agent AP is committed to through joint collaboration on the same artifacts. We can immediately see those agents with whom agent AP has collaborated most frequently, as well as the number of artifacts on which they have collaborated. This also indicates those agents with whom she has not yet collaborated. Alternatively, we could look at those agents that agent AP is committed to as a consequence of artifact-based RT relations to others' contributions, as suggested in the authorial trails we provided earlier.
Part IV: Validation

Table 23: Agent AP's agent commitment store.

<table>
<thead>
<tr>
<th>Agent committed to</th>
<th>Number of artifacts on which collaborate</th>
</tr>
</thead>
<tbody>
<tr>
<td>AD</td>
<td>6</td>
</tr>
<tr>
<td>AJ</td>
<td>6</td>
</tr>
<tr>
<td>AQ</td>
<td>4</td>
</tr>
<tr>
<td>AT</td>
<td>4</td>
</tr>
<tr>
<td>AU</td>
<td>4</td>
</tr>
<tr>
<td>AW</td>
<td>3</td>
</tr>
<tr>
<td>AV</td>
<td>3</td>
</tr>
<tr>
<td>AR</td>
<td>3</td>
</tr>
<tr>
<td>AX</td>
<td>3</td>
</tr>
<tr>
<td>AC</td>
<td>2</td>
</tr>
<tr>
<td>AE</td>
<td>2</td>
</tr>
<tr>
<td>AA</td>
<td>2</td>
</tr>
<tr>
<td>AI</td>
<td>1</td>
</tr>
<tr>
<td>AS</td>
<td>1</td>
</tr>
<tr>
<td>AB</td>
<td>1</td>
</tr>
</tbody>
</table>

The intersections and differences between agent commitment stores can uncover many interesting details. For example, we can:

(1) Examine which agents have collaborated with specified agents the most or least often.

(2) Identify which agents are committed to the same artifacts and for the same aspects.

(3) Identify which agents are committed to the same set of other agents.

(4) Identify which of the Company's agents have collaborated with its customers.

(5) Identify which of the Company's agents have contributed to artifacts which are the initial sources of requirements.
6.3. Personnel-Based RT and the Frequently Asked Questions

Having gathered the information necessary to model the contribution structure underlying the artifacts produced in the DR project, we now show how each of the frequently asked questions given in Part III of this thesis can be answered. Where there is an overlap in the answers to these questions, we try to illustrate the later answers using different material from the project.

6.3.1. FAQ 1: Involvement

"Who has been involved in the production of this requirement and how?"

Box 22: Question 1 of the frequently asked questions - Involvement.

Although we have not gone into the internal content of artifacts in the above account, we focus on a specific requirement that was in the requirements and design specification to illustrate how this question can be answered. One of the many requirements in version two of this specification, artifact 49, was to do with security. This requirement was mentioned in Section 1.1.4. of artifact 49 and its solution options were discussed in Sections 2.1.4. and 2.2.4. of the same artifact. This requirement was then pursued throughout the early phases of the project, was cited in all six versions of the requirements and design specification, but was later dropped in phase three. It was a requirement that led to a considerable amount of investigation and the production of artifacts that were later rendered redundant. It was under review for some time in phase three and, once removed, its subtle impacts only surfaced over time. It would have assisted matters to have been able to identify its original source and those who had pushed for its concern through the early artifacts. The provision of personnel-based RT could have enabled this.

In Figure 87, we can see that this requirement for security can be traced back to artifact 27. The contribution format at the source shows that agent AA was writing requirements in the name of the BB collective. If contribution formats had been provided for the internal components of this artifact, then agent AX could have been identified as the originator of this need. Without this link to its ultimate source, it would be hard to recover agent AX's original intention behind the issue he raised which, as it happens, was misconstrued by agent AA and never later rectified. We can see how this requirement pervades subsequent artifacts, due to agent AA's backing, though we only show the forwards impact to its first appearance in the second version of the requirements and design specification in Figure 87 for clarity. We can also see which agents ended up doing the most redundant work as a direct consequence of this requirement being introduced and pursued. Notably, it was a requirement which dominated a lot of AA's early and individual contributions in the project. Only agents AE and AC were also directly involved with this requirement before it became part of the requirements and design specification, and this was only when contributing in collaboration with agent AA.
Figure 87: Who was involved in the production of security requirements, a requirement first mentioned in a specific section of version two of the requirements and design specification, artifact 49. Note that we assume that relatively coarse relations have been put in place to link the internal components of the artifacts shown in the above diagram, which we have referred to within the boxes representing each artifact. Where groups of agents have contributed to an artifact, like BB in artifact 27, we also assume that the specific group members contributing to the internal components of these artifacts have also been delineated.
6.3.2. FAQ 2: Responsibility

"Who was originally responsible for this requirement, who is currently responsible for it, and at what points in its life has this responsibility changed hands?"

**Box 23: Question 2 of the frequently asked questions - Responsibility.**

To illustrate how this question can be answered, we do not focus on a requirement as such, but on a coarser artifact. Phase three of the DR project saw the introduction of a manager's guide to accompany the operations manual, its latest version in the project being artifact 160. Based on the information we have gathered, we show how we can determine who was originally responsible for the introduction of this guide, as well as who was involved in its subsequent evolution. Table 24 lays out the relevant information about the latest version of the guide in order to do this. Note that we could choose to selectively examine those agents who had been responsible for particular aspects of the guide, like for it's effect, content, appearance, and the like.

<table>
<thead>
<tr>
<th>Manager's guide</th>
<th>Artifact 160</th>
<th>Artifact 150</th>
<th>Artifact 138</th>
<th>Artifact 125</th>
<th>Artifact 118</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version number</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Principal</td>
<td>AJ</td>
<td>AJ</td>
<td>AJ</td>
<td>AI</td>
<td>AI</td>
</tr>
<tr>
<td>Author</td>
<td>AJ</td>
<td>AJ</td>
<td>AJ</td>
<td>AI</td>
<td>AI</td>
</tr>
<tr>
<td>Documentor</td>
<td>AM</td>
<td>AL</td>
<td>AL</td>
<td>AI</td>
<td>AI</td>
</tr>
<tr>
<td>Adopts relations</td>
<td>Adds to 150</td>
<td>Adds to 138</td>
<td>Adds to 125</td>
<td>Alters 118</td>
<td>None</td>
</tr>
<tr>
<td>References</td>
<td>Matched with 159</td>
<td>Matched with 149</td>
<td>Matched with 137</td>
<td>Matched with 124</td>
<td>(a)Matched with 115 (b)Framed by 108</td>
</tr>
<tr>
<td>Principal (of referenced artifact)</td>
<td>AJ</td>
<td>AJ</td>
<td>AJ</td>
<td>AI</td>
<td>(a)AI (b)AA</td>
</tr>
<tr>
<td>Author (of referenced artifact)</td>
<td>AJ</td>
<td>AJ</td>
<td>AJ</td>
<td>AI</td>
<td>(a)AI (b)AA</td>
</tr>
<tr>
<td>Documentor (of referenced artifact)</td>
<td>AM</td>
<td>AL</td>
<td>AL</td>
<td>AI</td>
<td>(a)AI (b)AA</td>
</tr>
</tbody>
</table>

**Table 24:** Changes in responsibility for the manager's guide.
Part IV: Validation

Table 24 shows the artifact-based RT relations between the different versions of the manager's guide, who contributed to the different versions, and any artifact-based RT relations to other project artifacts. From this information, we can clearly see the transition point between agent AI's original work on the manager's guide and agent AJ's later work on it. We can also see that agent AJ has only made additions to what agent AI originally produced and refined in version two. The working arrangements have also changed from agent AI working on his own as a true author, to agent AJ working as a devisor in conjunction with one other agent working as a relayer. To answer the question more specifically, we can see that agent AI was originally responsible for all aspects of the manager's guide, whilst agent AJ is now responsible for aspects such as its content and effect, and whilst agent AM is now responsible for aspects such as its appearance. Table 24 also shows that the manager's guide has been closely aligned with the various versions of the operations manual throughout its evolution, these being artifacts 150, 150, 137, 135, and 115. The only other artifact with which the very first version of the manager's guide is related is artifact 106. Inspection of this artifact can reveal that agent AA, as its author, was the agent who ultimately suggested having a manager's guide to accompany the operations manual.

6.3.3. FAQ 3: Working Arrangement

"At what points in this requirement's life have the working arrangements of all those involved been changed?"

**Box 24: Question 3 of the frequently asked questions - Working Arrangement.**

To illustrate how this question can be answered, we now focus on the formal versions of the requirements and design specification. In Table 25, we list its versions and the social roles of each of their contributors. From this information, we can see that the only consistency throughout the versions is that agent AE remains its sole documentor and one of its authors. We can see the subtle transformation in the role relation between agents AE and AA, through versions two and three, as other agents become involved. We can also see when the role relations between all involved became stabilised. From such information, if extended to consider the intermediate artifacts produced throughout this versioning process, we could begin to examine the impact of these changing working arrangements on different attributes of the requirements and design specification, like its quality, its quantity of technical material, and so forth.
Table 25: The working arrangements of those involved in the versions of the requirements and design specification.

6.3.4. FAQ 4: Change Notification

“Who needs to be involved in, or informed of, any changes proposed to this requirement?”

Box 25: Question 4 of the frequently asked questions - Change Notification.

Changes were not made to the content of the operations manual after agent AI left the project, so as of version three given in artifact 124. Only additions were subsequently made in its later versions, the latest addition in version six, artifact 159, being a new section describing how fax mailbox services fit into the generic DR package. If a change were now proposed to this latest section, we would be able to identify all those involved in the production of this section through personnel-based RT, as well as all those making subsequent use of it. Use of such personnel-based RT to assist the change process is shown in Figure 88.
Figure 88: Those agents that need to be involved in, and informed of, any changes proposed to the section on fax mailbox services in the operations manual, artifact 159. Note that artifacts 102 and 129 in this diagram were correspondence with potential customers, so we are able to trace back no further here. Artifacts 108 and 149 were versions of the operations manual, without the section on fax mailbox services, so we need not trace back any further here.
In examining those agents involved in the production path of the section of the operations manual on fax mailbox services, we can see that the service appears to have come about following correspondence between agent AJ and customer CF. However, we can also see that the requirement for a fax mailbox service was raised earlier in agent AI's list of requirements gathered from potential customers, artifact 114, where it was specifically raised by customer BX in correspondence. This requirement had subsequently been discussed at a meeting, documented in artifact 122, and had been researched further in artifact 127. In examining those agents involved in the usage path of the section on fax mailbox services, we can see that artifact 159 is adopted by artifact 166, and referenced by artifacts 160 and 161. If internal links were present from the fax mailbox service section in artifact 159 to artifact 166 we could see that we would need to inform agents AT and AW of any change. If we assume project policy is to inform the authors of any artifacts referencing ones which are to be changed, we could also see that we would need to inform agents AJ and AF. Using all such information that can be obtained through personnel-based RT, those project contributors to involve in or inform about change could be determined on a project-specific basis, and automatically flagged as changes get proposed or made. Such retrieval could be filtered to, say, inform only the documentors of those agents adopting the artifact to which a superficial change of wording is made, thus enabling different types of change to be dealt with in the most suitable way.

6.3.5. FAQ 5: Ramification

"What are the ramifications, with respect to the loss of possible requirements-related knowledge, if a specific individual or group leaves a project?"

Box 26: Question 5 of the frequently asked questions - Ramification.

To answer this question, we consider when agent AC left the DR project at the end of phase two. Before he left, we would need to examine agent AC's contributions to see which ones are known to the other key players in the project, which ones are only known to minor players, and which ones are not known to others. This is necessary to ensure that his contributions are passed on to relevant parties and not lost. In Table 26, we show agent AC's contributions and the other agents who also contributed to these artifacts where applicable. We also show those artifacts which adopt or reference each of agent AC's contributions to thereby examine their contributors in turn.
<table>
<thead>
<tr>
<th>Artifact</th>
<th>Social role of agent AC</th>
<th>Other contributors</th>
<th>Adopted by artifacts</th>
<th>Referenced by artifacts</th>
</tr>
</thead>
<tbody>
<tr>
<td>26</td>
<td>True author</td>
<td>No</td>
<td>30</td>
<td>28</td>
</tr>
<tr>
<td>28</td>
<td>Representative</td>
<td>AA = True author</td>
<td>30</td>
<td>29</td>
</tr>
<tr>
<td>30</td>
<td>True author</td>
<td>No</td>
<td>34/35/36</td>
<td>31/32/33/39</td>
</tr>
<tr>
<td>31</td>
<td>Devisor</td>
<td>AP = Relayer</td>
<td>None</td>
<td>32/33</td>
</tr>
<tr>
<td>34</td>
<td>Devisor</td>
<td>AA = Representative author AE/AG = Ghost author</td>
<td>None</td>
<td>47</td>
</tr>
<tr>
<td>35</td>
<td>True author</td>
<td>AA/AE/AG = Ghost author</td>
<td>None</td>
<td>50</td>
</tr>
<tr>
<td>36</td>
<td>Devisor</td>
<td>AA/AG = Ghost author AE = Representative author</td>
<td>45</td>
<td>44</td>
</tr>
<tr>
<td>37</td>
<td>Devisor</td>
<td>AA = Representative author AE/AG = Ghost author</td>
<td>None</td>
<td>47</td>
</tr>
<tr>
<td>38</td>
<td>Devisor</td>
<td>AA/AG = Ghost author AE = Representative author</td>
<td>None</td>
<td>44</td>
</tr>
<tr>
<td>50</td>
<td>True author</td>
<td>No</td>
<td>56</td>
<td>54/55/85</td>
</tr>
<tr>
<td>51</td>
<td>Devisor</td>
<td>AA = Representative author AE/AG = Ghost author</td>
<td>None</td>
<td>55</td>
</tr>
<tr>
<td>52</td>
<td>True author</td>
<td>AA/AE/AG = Ghost author</td>
<td>58</td>
<td>56</td>
</tr>
<tr>
<td>53</td>
<td>Devisor</td>
<td>AA/AG = Ghost author AE = Representative author</td>
<td>58</td>
<td>None</td>
</tr>
<tr>
<td>56</td>
<td>True author</td>
<td>AE = Ghost author</td>
<td>None</td>
<td>58/73/85</td>
</tr>
<tr>
<td>58</td>
<td>True author</td>
<td>AE = Ghost author</td>
<td>60/61</td>
<td>59</td>
</tr>
<tr>
<td>59</td>
<td>True author</td>
<td>AE = Ghost author</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>61</td>
<td>Devisor</td>
<td>AA = Ghost author AE = Representative author</td>
<td>63/64/65</td>
<td>None</td>
</tr>
<tr>
<td>63</td>
<td>Devisor</td>
<td>AA = Representative author AE/AG = Ghost author</td>
<td>None</td>
<td>68</td>
</tr>
<tr>
<td>64</td>
<td>True author</td>
<td>AA/AE/AG = Ghost author</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>65</td>
<td>Devisor</td>
<td>AA/AG = Ghost author AE = Representative author</td>
<td>74</td>
<td>None</td>
</tr>
<tr>
<td>73</td>
<td>True author</td>
<td>AT/AU = Ghost author</td>
<td>75/76/77</td>
<td>80/87</td>
</tr>
<tr>
<td>74</td>
<td>Devisor</td>
<td>AA/AG = Ghost author AE = Representative author</td>
<td>75/76/77</td>
<td>None</td>
</tr>
<tr>
<td>75</td>
<td>Devisor</td>
<td>AA = Representative author AE/AG = Ghost author</td>
<td>None</td>
<td>78/79</td>
</tr>
</tbody>
</table>
### Table 26: Agent AC's contributions, collaborators, and the subsequent project artifacts which have made use of his contributions in some way.

Note that if agent AC is himself a contributor to the artifacts cited in columns four and five of this table, then the artifact identifier is given in bold font. We can thereby see which artifacts are not made use of by distinct others to clarify which may not be known to others.

By inspection of Table 26, we can see that agent AE is aware of agent AC's individual contribution in artifact 80, as he adopted its content in artifact 81 when working in conjunction with agent AC. However, we can see that artifacts 85, 86, and 87 may not be known by the other key players, since agents AQ, AP, and AT are relatively minor players in the phase concerned, so would signal the need to ensure agent AC's responsibilities were handed over before he left the project. Furthermore, we can see which agents have contributed with agent AC, and in what role relations, to pass on this information if there are later queries about the artifacts concerned. If a new agent is to take over agent AC's responsibilities, this agent can further identify agent AC's long-term and transitory collaborators for contact purposes. In addition, by indicating those agents who have made use of agent AC's contributions, especially in conjunction with AC himself, we can signal those agents who are most likely to have had additional communication with agent AC about his contributions at some stage. Furthermore, before agent AC leaves the project, we can signal which of his artifacts are still pending approval for integration into the critical path of the project.

<table>
<thead>
<tr>
<th>Artifact (cont...)</th>
<th>Social role of agent AC (cont...)</th>
<th>Other contributors (cont...)</th>
<th>Adopted by artifacts (cont...)</th>
<th>Referenced by artifacts (cont...)</th>
</tr>
</thead>
<tbody>
<tr>
<td>76</td>
<td>True author</td>
<td>AA/AE/AG = Ghost author</td>
<td>81</td>
<td>78/80</td>
</tr>
<tr>
<td>77</td>
<td>Devisor</td>
<td>AA/AG = Ghost author</td>
<td>81/83/84</td>
<td>78</td>
</tr>
<tr>
<td>78</td>
<td>Devisor</td>
<td>AA/AG = Ghost author</td>
<td>82</td>
<td>81</td>
</tr>
<tr>
<td>80</td>
<td>True author</td>
<td>None</td>
<td>81</td>
<td>85</td>
</tr>
<tr>
<td>81</td>
<td>True author</td>
<td>AE = Representative author</td>
<td>83/84</td>
<td>82/85</td>
</tr>
<tr>
<td>82</td>
<td>Devisor</td>
<td>AE = Representative author</td>
<td>None</td>
<td>84</td>
</tr>
<tr>
<td>84</td>
<td>Devisor</td>
<td>AA/AG = Ghost author</td>
<td>88</td>
<td>None</td>
</tr>
<tr>
<td>85</td>
<td>Nominal author</td>
<td>AQ/AP = Ghost author</td>
<td>None</td>
<td>86</td>
</tr>
<tr>
<td>86</td>
<td>True author</td>
<td>AQ = Ghost author</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>87</td>
<td>True author</td>
<td>AT = Ghost author</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>88</td>
<td>Devisor</td>
<td>AA/AG = Ghost author</td>
<td>None</td>
<td>None</td>
</tr>
</tbody>
</table>

*Table 26: Agent AC's contributions, collaborators, and the subsequent project artifacts which have made use of his contributions in some way. Note that if agent AC is himself a contributor to the artifacts cited in columns four and five of this table, then the artifact identifier is given in bold font. We can thereby see which artifacts are not made use of by distinct others to clarify which may not be known to others.*
6.3.6. FAQ 6: Source

"To whom should I refer for more information about this requirement?"

Box 27: Question 6 of the frequently asked questions - Source.

In version six of the operations manual, artifact 159, there is a section on call divert services. To identify those agents able to provide more information about this service, we can identify when this service was introduced as a new section in the operations manual, uncover the trace of related artifacts prior to its introduction, and examine whether any changes have since been introduced. In the following description and figure, we again indicate how relations between coarse-grained internal components of artifacts can help direct the personnel-based RT more specifically.

Figure 89 shows that there are two relations from artifact 159 to artifacts 158 and 149. Artifact 158 has no internal links to the call divert service of artifact 159, so need not be traced back further. Whereas, artifact 149 is version five of the operations manual with internal links between their sections on the call divert service, so needs to be traced back further. Artifact 149 has two relations to artifacts 137 and 147. As 137 is version four of the operations manual without the section on the call divert service, this need not be traced back further. Whereas, artifact 147 is directly copied into artifact 149 to describe the call divert service, so needs to be traced back further. Artifact 147 has two relations to artifacts 145 and 146. Artifact 147 adds to a draft description of the service, documented in artifact 145, and this addition is substantiated by informal notes in artifact 146. Tracing artifact 145 back further, we can see that it is matched with artifact 137, an earlier version of the operations manual to guide its structure and content. We can also see that it was caused by correspondence with a specific customer and that it was further substantiated by research looking into supporting this customer's requirement in artifact 142.
Figure 89: Those agents that can provide more information about the call divert service.

To answer the question regarding who to refer to for more information, although agents AJ and CC, the latter a member of BM, could inform about the original requirement for the call divert service, agents AE and AJ could inform more about its details. Each of the members of the BE group could also probably inform about various aspects of the service. Furthermore, this points out that agent AE is likely to be a sufficient contact point now that agent AJ has left the project. However, the ability to identify agent AJ’s contributions without agent AE before he left the project would have put agents AJ and AE in a better position to uncover and smooth over any outstanding issues together beforehand.

6.3.7. FAQ 7: Remit

"Within the remit of which group do decisions about this requirement lie?"

Box 28: Question 7 of the frequently asked questions - Remit.

Through Figure 90, we can see that any decisions about the later versions of the requirements and
design specification produced in phase two of the DR project lay with the group involved, namely agents AC, AA, AE, and AG. However, decisions for its earlier versions lay with different subsets of this group at various times. What is interesting here is how the ultimate responsibility for the requirements and design specification changes amongst these agents over time. Once agent AC left the project, after version six had been produced, it would have been interesting if its ultimate responsibility were seen to rotate back to one of those who had been its principal earlier on. Instead, its ultimate responsibility passed on to agent AH. After this had happened, the joint contributions by the remaining group members, agents AE, AA, and AG, were not sustained. In fact, and by looking at our data, we can see that they never did all come together again to contribute once agent AH was project leader. However, they came together one more time when agent AI took over as project leader, which probably reflects why subsequent development from this requirements and design specification proceeded more successfully the second time around.

![Diagram](image)

**Figure 90:** The changing decision making authority amongst members of the key group contributing to the requirements and design specification.

### 6.3.8. FAQ 8: Copying

"Who was responsible for copying this piece of information into this requirement?"

**Box 29:** Question 8 of the frequently asked questions - Copying.

It would be a simple exercise to obtain the information needed to answer this question. For instance, say we were interested in finding out who copied the section on freephone services into the operations manual, or the section on call divert services, or the section on fax mailbox services. We can use the adopts artifact-based RT relations to locate those versions of the operations manual in which these sections were first added, a simple task if coarse relations are maintained amongst equivalent sections of the operations manual in its different versions. We can then examine the artifacts surrounding the version where the addition happened, to determine whether the section in
question had been directly copied into the operations manual from a separate artifact, such circumstances indicating direct transcription. We can then read off the documentors to locate those responsible. By doing exactly this with the DR data, we can see that:

(1) Agent AL was responsible for copying the information on the freephone service into version four of the operations manual, artifact 137, having copied agent AW's written notes given in artifact 135.

(2) Agent AL was responsible for copying the information on the call divert service into version five of the operations manual, artifact 149, having copied agent AJ's notes given in artifact 147.

(3) Agent AM was responsible for copying the information on the fax mailbox service into version six of the operations manual, artifact 159, having copied agent AJ's notes given in artifact 158.

6.3.9. FAQ 9: Use

"Who uses this requirement and how?"

**Box 30: Question 9 of the frequently asked questions - Use.**

Again, it would be a simple exercise to retrieve the information needed to answer this question. As an example, say we were interested in finding out who used agent AG's marketing plan and how. We can determine which artifacts used this plan by examining its incoming adopts and references relations and reading off the agents that contributed to those artifacts found connected to it. These are the agents related to agent AG through artifact-based RT relations, as opposed to those related through collaboration on the same artifact. By looking at Figure 91, we can see that the marketing plan, artifact 57, has only ever been adopted by agent AG. We can also see that it has mainly been referenced by agent AG when contributing alone or involved as one of the contributors. Only once has it been used as a reference point by an artifact to which agent AG had not contributed. This was by agent AI when initially drawing together existing artifacts at the outset of the phase in which he took over as project leader. We can therefore see that this artifact has not been widely used by the project or its participants.
6.3.10. FAQ 10: Collaboration

"On which requirements do certain agents collaborate together and how?"

Box 31: Question 10 of the frequently asked questions - Collaboration.

In Tables 27 through to 29, we list those artifacts on which agents AE and AA have collaborated in the different phases of the DR project, to show how this question can be answered. We also indicate the social roles in which they contributed, as well as the social roles of other joint contributors. An analysis of their different contributions is then summarised in Tables 30 and 31.

<table>
<thead>
<tr>
<th>Artifact</th>
<th>AE's social role</th>
<th>AA's social role</th>
<th>Others' social roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>3,8,11,17</td>
<td>Ghost author</td>
<td>True author</td>
<td>None</td>
</tr>
<tr>
<td>4</td>
<td>Ghost author</td>
<td>Devisor</td>
<td>AU = Relayer</td>
</tr>
<tr>
<td>16</td>
<td>Devisor</td>
<td>True author</td>
<td>BI = Devisor</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BJ = Ghost author</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BK = Ghost author</td>
</tr>
<tr>
<td>19</td>
<td>Ghost author</td>
<td>True author</td>
<td>AT = Ghost author</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AU = Ghost author</td>
</tr>
<tr>
<td>22</td>
<td>Sponsor</td>
<td>Devisor</td>
<td>AP = Relayer</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AB = Sponsor</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BF = Sponsor</td>
</tr>
</tbody>
</table>

Table 27: The collaborations of agents AE and AA in phase one of the DR project.
### Part IV: Validation

<table>
<thead>
<tr>
<th>Artifact</th>
<th>AE’s social role</th>
<th>AA’s social role</th>
<th>Others’ social roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>24</td>
<td>Ghost author</td>
<td>True author</td>
<td>None</td>
</tr>
<tr>
<td>34,37,51,63,75</td>
<td>Ghost author</td>
<td>Representative author</td>
<td>AG = Ghost author</td>
</tr>
<tr>
<td>35,52,64,76</td>
<td>Ghost author</td>
<td>Ghost author</td>
<td>AC = True author</td>
</tr>
<tr>
<td>36,38,53,65,74,77,78,84,88</td>
<td>Representative author</td>
<td>Ghost author</td>
<td>AG = Ghost author</td>
</tr>
<tr>
<td>48,49</td>
<td>Representative author</td>
<td>Devisor</td>
<td>None</td>
</tr>
<tr>
<td>61</td>
<td>Representative author</td>
<td>Ghost author</td>
<td>AC = Devisor</td>
</tr>
</tbody>
</table>

**Table 28**: The collaborations of agents AE and AA in phase two of the DR project.

<table>
<thead>
<tr>
<th>Artifact</th>
<th>AE’s social role</th>
<th>AA’s social role</th>
<th>Others’ social roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>93</td>
<td>Ghost author</td>
<td>Representative author</td>
<td>AH = Devisor</td>
</tr>
<tr>
<td>94</td>
<td>Representative author</td>
<td>Ghost author</td>
<td>AH = Devisor</td>
</tr>
<tr>
<td>95</td>
<td>Ghost author</td>
<td>Ghost author</td>
<td>AH = True author</td>
</tr>
<tr>
<td>96</td>
<td>Representative author</td>
<td>Ghost author</td>
<td>AH = Sponsor</td>
</tr>
<tr>
<td>99</td>
<td>Ghost author</td>
<td>Ghost author</td>
<td>AI = Nominal author</td>
</tr>
<tr>
<td>105</td>
<td>Ghost author</td>
<td>Ghost author</td>
<td>AI = True author</td>
</tr>
<tr>
<td>122</td>
<td>Ghost author</td>
<td>Ghost author</td>
<td>AI = True author</td>
</tr>
</tbody>
</table>

**Table 29**: The collaborations of agents AE and AA in phase three of the DR project.
From all these tables, we can see that early on in the DR project, agent AA had more ultimate responsibility for the project than agent AE and was acting in more of the three capacities. This ultimate responsibility diminished significantly in phase two, totally disappeared in phase three, and was not transferred to agent AE. We can see that agent AE has always been an author in conjunction with agent AA when they collaborated. What is interesting is that agent AA did most of their joint documentation in phase one, agent AE took over a lot of this responsibility in phase two, and then both agents AA and AE did little documenting when they collaborated together in phase three, implying a third-party took over that aspect.

In phase one when agents AA and AE collaborated, only the two of them were involved in exactly half of their jointly contributed artifacts. In phase two, very few artifacts were contributed by agents AA and AE on their own. In phase three, agents AA and AE always collaborated in conjunction with one of the project leaders, first agent AH, and then agent AI. This may indicate that agents AE and AA found that one-to-one collaboration was most needed early on in the project until others got involved and eventually took over. Interestingly, agents AA and AE collaborated no further once agent AJ took over as project leader. Perhaps this indicated that what was passed on to agent AJ did not warrant their further interaction, leaving them free to instigate other projects.
6.4. Further Analyses

It is interesting to point out some of the information that becomes available as a by-product of the approach, like information about the contributor and contribution profiles of a project. This is the objective of this sub-section.

6.4.1. Number of Contributors to Artifacts

Figures 92 to 95 show the number of contributors to the artifacts produced in the four phases of the DR project and, in so doing, illustrates the contribution profiles for the different phases. The information in these figures can be used to identify those artifacts on which many agents have contributed, perhaps indicating those artifacts more prone to later query, as well as to identify those artifacts on which a single agent has contributed.

Figure 92 shows the contribution profile for the start-up phase of the DR project. We can see an early peak in the number of contributors, followed by a falling off in number as certain agents drive the project forward and others drop out having given their input. It appears to be a phase with lots of regular group contributions, interspersed by a number of artifacts produced by smaller groups of agents, eventually leading back into larger group contributions.

Figure 93 shows the contribution profile for the specification phase of the DR project. We can see a few larger group contributions at either end, with regular contributions by four agents in between, these being interspersed by contributions from individuals and pairs of agents. It appears

---

**Figure 92:** The number of contributors to the artifacts produced in phase one of the DR project.
to be a phase that is more structured than the first.

Figure 93: The number of contributors to the artifacts produced in phase two of the DR project.

Figure 94 shows the contribution profile for the design and development phases of the DR project. As agent AH was project leader for artifacts 89 to 98, and agent AI was project leader for artifacts 99 to 127, we consider this figure in these two parts. In part one led by agent AH we can see that, not only were few artifacts produced but, on average, there was generally two or more agents contributing to these. No group contributions with over three agents took place at the beginning of this part. Such group collaboration only occurred once and towards the end of this part. This indicates that agent AH relied on other agents being involved and that there was probably not a sufficient change-over of ideas between all when agent AH took over. In contrast, and in the part led by agent AI, we can see that it starts with two large group contributions. The majority of the subsequent artifacts were then produced by an individual, though groups of agents got involved periodically. The profile of this part is mainly flat with random spurts of group involvement.
Figure 94: The number of contributors to the artifacts produced in phase three of the DR project.

Figure 95 shows the contribution profile for the refinement and maintenance phases of the DR project. It is also discussed in two parts, to differentiate when agent AJ was project leader for artifacts 128 to 162, from when agent AK was project leader for artifacts 163 to 166. We can see that the contribution profile for the first part appears very similar to that of part two in Figure 94, but with small groups of contributors replacing the individual contributors. In contrast, we can see that the second part starts off in a similar way to part one in Figure 94, so agent AK does not yet appear to interacting with many agents to further the DR service. How such staff transitions are handled are critical to keep the momentum of the project going. Although there could be many reasons for this, the lack of any group contributions could signal that the service has stabilised, or else that there could be too many assumptions that things are running smoothly.
In Figure 96, we concatenate the previous four figures to give the contribution profile for the entire DR project. We can see that the different phases do appear to stand out. The number of contributors at first appears very spiky, then settles with the transition from the creative and informal phase to the more structured phase intent on producing a baseline requirements and design specification from which to develop the DR service. This is followed by a phase with consistent involvement by solitary agents, interspersed with a few group efforts, a profile then repeated but with consistent involvement by small groups of agents instead of individuals. Although premature to generalise, interesting future work could be to consider the health of a project in terms of its contribution profiles in its various phases.
Figure 96: The number of contributors to all the artifacts produced in the DR project. This is the project's contribution profile.

6.4.2. Number of Contributors to Artifacts in the Three Capacities

The previous figures can each be repeated to show the number of contributors to the artifacts produced in each of the three capacities, but is only demonstrated for phase one in Figure 97. This further information can be used to locate those artifacts on which many agents have contributed as principal, author, or documentor, and those artifacts on which a single agent has contributed in these capacities. Again, this information can be used to examine the contribution profiles with respect to each capacity throughout a project. Where there are large numbers of documentors, warnings about potential errors in transcription could be signalled, and so forth.
In Table 32, we summarise this contributor information for the whole DR project, where we can see that the average number of contributors to the artifacts produced is 2.6. It is interesting that it appears more common to have larger numbers of collaborating authors, as opposed to principals and documentors. If this situation were indicative of most projects, it would be useful to examine how the notion of "author" as we have defined it could be more finely decomposed to capture, say, the degree of an agent's contribution as author.
Table 32: Summary information about project contributors. Note that the maximum number of contributors in this table is really a consequence of how we have chosen to analyse our data, since we grouped all customer correspondence into a single artifact for simplicity. If we had not done this, the maximum number of principals, authors and documentors would have been less.

6.4.3. Number and Type of Contributions by Specific Agents

In Figure 98, we show the total number of contributions that different agents have made throughout the course of the DR project. This illustrates the contributor profile for the project. In Figures 99 through to 101, we give this information in terms of the total number of artifacts different agents have contributed as principal, author, or documentor throughout the course of the project. Similarly, these give the principal, author, and documentor profiles for a project. We repeat each of these graphs for the four project phases in Appendix D.5.
Part IV: Validation

Figure 98: The total number of artifacts different agents have contributed to in the DR project. Note that we omit one-off contributions for clarity.

Figure 99: The total number of artifacts different agents have contributed to as principal in the DR project. Note that we omit one-off contributions for clarity.
Figure 100: The total number of artifacts different agents have contributed to as author in the DR project. Note that we omit one-off contributions for clarity.

Figure 101: The total number of artifacts different agents have contributed to as documentor in the DR project. Note that we omit one-off contributions for clarity.
The above figures show that agents AE and AA are the driving force of the project. Surprisingly, although agent AC was only involved in phase two, he can be seen to have held a great deal of responsibility when developing the baseline specification for the generic DR service. In addition, he contributed more tangible artifacts to the project than agent AD who took over his company position and held his responsibilities for a lot longer. We can also see that agent AI's and agent AJ's respective contributions as project leader were far greater than those of agents AH and AK when occupying the same company position. Furthermore, agent AP and members of the group BB can be seen to have provided a stable backbone for the project, whilst customers BH, BX, CC, and CF can be seen to have been the most influential in shaping the eventual service. If we examine the same figures for each of the project phases, given in Appendix D.5., we can further examine the driving force and stable backbone for each project phase.

Just as we have pointed to the leverage that can be gained by looking into the contribution profiles for a project, contributor profiles can similarly provide the scope for many forms of analysis. An agent's overall role in each phase of a project, as well as overall, could be revealed through such analyses. We could then use such information to identify which agents in an organisation are best suited to different phases of a project. As an example, we summarise this kind of information for agents AE and AA in Figures 102 and 103. Their contributions in the three capacities, as a percentage of their total contributions, is then further summarised in Figures 104 and 105.

![Figure 102: The contributor profile of agent AE.](image-url)
Part IV: Validation

**Figure 103:** The contributor profile of agent AA.

**Figure 104:** The percentage of agent AE’s contributions in each of the three capacities.
6.4.4. Key Contributors

In Table 33, we list the key contributors to the DR project, determined according to the total number of artifacts that agents contributed to throughout. We then indicate the project phases in which these key contributors contributed in Table 34. What we can see from these tables is that, of the seven highest contributors, only agent AE has contributed in all four of the project phases. We can also see that the lower contributors, most being members of the BB group, have been steady contributors throughout. Another interesting point is that agents AC, AJ and AI, although key contributors, have each only contributed in one of the four project phases, as each acted as the other's replacement in the project.

<table>
<thead>
<tr>
<th>Number of artifacts contributed to</th>
<th>Agents</th>
</tr>
</thead>
<tbody>
<tr>
<td>35+ artifacts</td>
<td>AE/AA</td>
</tr>
<tr>
<td>21 to 34 artifacts</td>
<td>AC/AJ/AG/Al</td>
</tr>
<tr>
<td>11 to 20 artifacts</td>
<td>AD/AT/AV/Al</td>
</tr>
</tbody>
</table>

*Table 33: The key contributors to the DR project.*


<table>
<thead>
<tr>
<th>Contributor</th>
<th>Phase 1</th>
<th>Phase 2</th>
<th>Phase 3</th>
<th>Phase 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>AA</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✘</td>
</tr>
<tr>
<td>AC</td>
<td>✘</td>
<td>✔</td>
<td>✘</td>
<td>✘</td>
</tr>
<tr>
<td>AJ</td>
<td>✘</td>
<td>✘</td>
<td>✔</td>
<td>✘</td>
</tr>
<tr>
<td>AG</td>
<td>✘</td>
<td>✔</td>
<td>✔</td>
<td>✘</td>
</tr>
<tr>
<td>AI</td>
<td>✘</td>
<td>✘</td>
<td>✔</td>
<td>✘</td>
</tr>
<tr>
<td>AD</td>
<td>✘</td>
<td>✘</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>AT</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>AV</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>AP</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>AR</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>AW</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>AU</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

Table 34: The phases of the DR project in which the key contributors have contributed.

If we examine the sort of information that we can obtain about these key contributors further, Table 35 indicates which of these agents collaborated together the most often. If we assume that frequent collaboration means that agents work well together, then we can see that agents AA and AE probably work well together. Due to two clusters shown in the top left and bottom right quadrants of this table, we can also see that the interaction between the highest contributors and members of the BB group who have regularly contributed, namely agents AT, AV, AR, AW, and AU, is not frequent. Perhaps the project could have benefited from more collaboration between the highest contributors and a representative from the BB group. What is also interesting is to examine the relations between two agents when then do collaborate together. For instance, agents AG and AV collaborate quite frequently together, but only ever in a devisor/relayer role relation. Such information could have signalled the extra demands that agent AG was placing on key members of the BB group prior to getting his own personal secretary much later.
Taking this information further, Table 36 indicates which of the key players are likely to be team players, which work mainly on a one-to-one basis, and which work predominantly on their own. We can see that, of the two most productive project leaders, agents AJ and AI, agent AI works mainly on his own, whilst agent AJ works mainly in small groups. Of the two managing directors, agents AC and AD, we can see that agent AC is a larger team player than agent AD, perhaps reflecting that he was a more suitable agent to ensure early project meetings and momentum. Agent AA, being the instigator of many of the larger collaborations, may also hold a clue as to why such meetings are notably absent in the later phases of the project after he had withdrawn.
6.5. Discussion

In the course of conducting this case study, a number of issues arose concerning both the use of the approach and the use of the information it provides. We list the most central of these below and then summarise what we found to be the main implications of the approach for practice.

6.5.1. Issues

The main issues that arose concerning the use of the approach included:

(1) Determining the finer semantics of artifact-based RT relations. We found that the key contributors had no problem when retrospectively putting in the main artifact-based RT relations between the artifacts produced in the project. Although we also found that there was consensus as to which of these relations were adopting or referencing in function, we cannot generalise to assume that this would be the case whilst practitioners were actively engaged in a project consisting of many more artifacts. In contrast, we found increasingly less consensus as finer semantics were given to these relations, particularly since the finer coherence and cohesion relations were not considered to be mutually exclusive. However, as the approach is mainly designed to operate with knowledge of the minimal semantics, the

<table>
<thead>
<tr>
<th>Contributor</th>
<th>On own</th>
<th>With 1 other</th>
<th>With 2 others</th>
<th>With 3+ others</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>13</td>
<td>22</td>
<td>7</td>
<td>25</td>
</tr>
<tr>
<td>AA</td>
<td>13</td>
<td>12</td>
<td>6</td>
<td>29</td>
</tr>
<tr>
<td>AC</td>
<td>4</td>
<td>9</td>
<td>3</td>
<td>29</td>
</tr>
<tr>
<td>AJ</td>
<td>2</td>
<td>12</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>AG</td>
<td>0</td>
<td>5</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>AI</td>
<td>13</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>AD</td>
<td>1</td>
<td>2</td>
<td>9</td>
<td>4</td>
</tr>
<tr>
<td>AT</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>AV</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>AP</td>
<td>0</td>
<td>2</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>AR</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>AW</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>AU</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>11</td>
</tr>
</tbody>
</table>

*Table 36: With how many other agents key agents collaborated when they contributed to the DR project.*
above situation would only have restricted some of the finer-grained analyses that could have been made possible as a by-product. The practitioners we dealt with agreed that it was probably most appropriate for artifact authors to establish the main relations between those artifacts they reference or adopt and to provide their minimal semantics, but for a dedicated agent to liaise with the authors if finer-grained semantics were required, so as to maintain consistency.

(2) Accounting for how an author actually contributes. We found that the key contributors were also able to retrospectively partition the contributors we had listed into principals, authors, and documentors when provided with some contextual information. However, we found that the capacity of "author" was still too broad, as often used to record all those involved in a meeting which generated an artifact. For example, agent AG appeared as a key author in the DR project, since he was present at many project meetings, although his contributions in these meetings were often minimal and negative. Perhaps such issues would have been dealt with if we had delineated the authors of the internal components in these artifacts to suggest the degree of an author's contribution. Concerns were also expressed by practitioners about who would collect the information about artifact contributors, since the contributor lists they worked from when partitioning the contributors into the three capacities had been collected by an outsider and contained information that could no longer be ascertained from the artifacts themselves. The consensus here seemed to be that such information should be determined and recorded by the documentor of each artifact, or preferably by a specific agent charged with this responsibility as part of project policy.

(3) The importance of artifact distribution details. An observation was made by one of the practitioners concerning the information we had retained about those agents to whom artifacts were either copied or passed on to throughout the DR project. By considering this information, in conjunction with that provided by the approach, we would be in a position to examine those contributions resulting from the distribution of other artifacts. As an example in a large project, copying artifacts to numerous individuals who never respond as a consequence could lead to a considerable waste of resources. Distribution information could be used to examine whether those to whom artifacts have been copied then actively contribute to the project as a consequence. This would reveal those agents that would need to be kept in artifact circulation loops or those to circulate to more often than others. Much of the extra information that would be needed to provide this potential could be captured automatically from the circulation lists that are often appended to artifacts.

(4) Indicating the purpose or topic of the artifacts. In analysing the information the approach made available, an observation was made concerning the nature of the various artifacts themselves. Some were found to be technical in nature, whilst others were mainly to do
with marketing, billing, new ideas, customers, and so forth. We found that, by giving some high-level details about the kind of information held in artifacts, we were in a position to examine the different threads running through the project's artifacts, and so to identify those agents most active in the different threads as a consequence. For example, we were able to see: (a) agent AE’s high profile across all the technically-related artifacts; (b) agent AA’s high profile across all those artifacts filled with new ideas; and (c) agent AD’s high profile as a principal throughout the later sales-related and contractual artifacts.

5) **Capturing the undocumented events leading up to an artifact.** Throughout the phase of the DR project that we followed most closely, we noted that a fair amount of informal interaction took place amongst the project contributors which did not result in project artifacts being directly produced, though much of this had a direct bearing on some artifacts that were produced. This perhaps signals the need to record the existence of such events, even if their content is omitted, so they can be integrated into the approach. However, as there are likely to be no boundaries when identifying those events that could be said to impinge on project artifacts, we first suggest determining what is lost by not recording such information to assess at what point this becomes significant for the viability of the approach.

The main issues that arose concerning the use of the information the approach makes available included:

1) **The lack of time to both analyse the data and act upon it.** The key practitioners in our case study stressed that, whilst it may not be too much trouble to record the minimal artifact-based RT relations and artifact contributors, they simply would not have the time to analyse the data the approach provides and then to act upon it. This indicates that the personnel-based RT, and the further analyses that the approach makes possible, would need to be made a more integral part of the approach.

2) **The overwhelming analytical opportunities.** As we have illustrated in the results of the case study, the information that the approach provides can be used to answer important questions which can alleviate RT problems. However, we have also shown how this information provides the potential for much more. This indicates that it would be crucial to examine in more detail that further information which can really inform practice in particular organisations or projects, either immediately or in the long-term, as opposed to simply being that which is interesting to know.

3) **Care in analysis and generalisation.** There is a need to be wary of some of the further analyses that can be carried out with the information the approach provides. For instance, the number of tangible artifacts that a specific agent has produced, whilst perhaps an indicator of their productivity, is not necessarily an indicator of the quality of their
contributions and of their centrality to the project. There are many such outstanding issues that would need to be explored in more detail here.

(4) **Sensitivity of information.** Surprisingly, the sensitive nature of the information the approach provides was not mentioned by the practitioners in our case study. However, since it was imperative to change the identities of the project contributors in the description of this case study, we believe it is an important issue. To address this issue, policies would need to be put in place to guide its use.

(5) **Advances over current practice.** Unfortunately, we were not able to quantitatively compare the benefits in practice of being able to trace contributors with the situation where this is not possible. This was due to the Company's lack of RT policy and the retrospective manner in which we applied the approach. This is a further outstanding issue that would need to be investigated.

### 6.5.2. Implications for Practice

The DR project was probably not the ideal project for demonstrating and evaluating our approach. The Company was probably not the most receptive to what the approach was trying to do, since RT was not practiced, and its systems and software development philosophy was rather informal and unstructured. However, we were fortunate in having access to such high quality data. A different perspective on the approach and its results would likely have been obtained by those organisations: (a) with some form of RT or document control already in place; (b) examining how to achieve process improvement; (c) currently experiencing RT problems; or (d) running larger projects with many people and documents involved.

Despite the above comments, members of the Company agreed that the data we revealed about the contribution structure underlying the DR project rang true, and that it would have pointed to the right agents where problems of misunderstanding and the like surfaced. In particular, they were surprised to see how extensive some of the trails of contributors behind their artifacts had actually been, though recognised how useful it could be to retrieve such information when considering change. It highlighted problems arising from the earlier work in this project that were only recognised later on, especially related to the changing staff members. It further provided information about agent social roles and role relations that could not have been determined from the Company's organisational chart or from the work allocation timetables for the project. This information was considered important to inform how work could probably be better allocated amongst agents in future projects. Even for a company not concerned with RT, the approach was considered a feasible addition to practice, whilst the personnel-based RT and value-added information it provides was considered likely to impact practice in a positive way.
As a consequence of this case study, we believe that the approach is probably best introduced in those organisations which are concerned with RT issues and already practice some form of RT. Moreover, that it would need to be introduced in an incremental manner, showing first what can be gained immediately from small changes in current practice before taking any steps further. With minimal extensions, we could provide the ability to selectively trace those agents who have been involved in different aspects of a project, and so show the benefits of a more comprehensive approach to RT. We could then demonstrate the ability to examine the working relations of those involved in a project, and so show how such information could be used to inform practice. We could go even further to demonstrate how such information, when gathered from across projects, could be used to examine how the organisation's development and RE processes could themselves be improved.

7. Critical Evaluation

In this section, we provide a critical evaluation of the approach, based upon the above scenario and experience with the case study. One point that is important to keep in mind when we evaluate the approach is that the manner in which the artifact-based RT relations and contribution formats are to be obtained remains something that can be determined on a project-specific basis. This means that application of the approach could be the distributed responsibility of all the agents involved in a project, the responsibility of a particular agent contributing to each of the artifacts produced, or the responsibility of a dedicated individual or group. Therefore, where organisational policies are already in place as to how artifact-based RT is to be established and maintained, we anticipate that our extensions would be handled in the same manner.

7.1. Evaluation Framework

As a basis for our evaluation, we adopt some of the challenges set out by [Grudin 1994], originally for the design and evaluation of groupware. These are criteria which we believe are important to examine:

(1) Does application of the approach require additional work from agents who do not directly benefit as a consequence?

(2) Must the approach be adopted by a high percentage of those involved if it is to be useful?

(3) Does the approach make allowance for the situated nature of the work involved?

(4) Does the approach threaten existing social, political, economic, or motivational factors that operate amongst those involved?

(5) Does the approach integrate with those applications commonly used to support the primary
activities of systems and software development?

(6) Were potential users involved in the development of the approach?

(7) What strategies need to be employed to encourage the uptake and acceptance of the approach in practice?

(8) Are we able to generalise from evaluation of the approach?

7.2. Balance Between the Work Involved and Benefits Reaped

The different issues confronting the providers and end-users of RT is one we have raised many times in this thesis. Whatever policy is chosen for collecting the information the approach requires, there is likely to be a disparity between the work involved by the providers and the benefits reaped by the end-users, since not all providers would also be end-users. We have suggested ways to reduce the manual effort necessary to collect the required information, like the inference of default contributors, but this is unlikely to eliminate the need for agent involvement. The potential disparity is probably best handled by introducing the dedicated job of an RT facilitator or by extending an existing job description to cover the responsibilities of provision. For such an individual or group, the work involved would no doubt outweigh the personal benefits reaped from personnel-based RT, though bring benefits to the larger collective. Those agents that would now be in a position to perform personnel-based RT to assist their working practices, those involved in the various phases of a project, would be the primary and direct beneficiaries of the approach. However, as the need for personnel-based RT is likely to be low at the onset of a project and steadily increase as it progresses, those agents involved early on are likely to get fewer immediate benefits and would be making potentially more work for themselves later on if they were to cooperate with the RT facilitator. To address such issues, policies would need to be put in place to stress how the information the approach provides is to be used, for how long, and so forth. Project leaders and managers would be indirect beneficiaries of the information the approach provides, since they would be in a position to learn something about the social structures underlying their projects. It is exactly these agents that would do little, if any of the work, and would potentially reap lots of value-added information.

7.3. Dependence on a Critical Mass

Were provision of the information the approach requires the joint responsibility of all involved, or the responsibility of one of the agents in the contribution format of each artifact, the approach would still enable a minimal form of personnel-based RT if only some of these agents complied. Although this would lead to missing contributor details at periods throughout a requirement's life, the minimal information could still guide the search process and so not cause the approach to fail
totally in its main objective. However, some of the value-added information provided would not be so reliable. As mentioned above, cooperation would be essential in the early phases of a project to prepare the groundwork for later personnel-based RT but, unless more immediate benefits were provided to these agents, the approach could be abandoned by the majority before the real benefits were reaped. Even if provision of this information were the responsibility of a dedicated individual or group, these agents would still depend on some cooperation from all those involved to gather the right information in a timely manner. This problem could be addressed by introducing a project "champion" charged to ensure support for the approach in the early phases. In contrast, few agents need to make use of the personnel-based RT that the approach makes possible for it to have been worthwhile. This is because the time saved in identifying those agents that need to be involved in even one change proposal could quite easily be offset against the time taken to establish the potential for this traceability in the first place.

### 7.4. Handling of Exceptions

The schemes we have suggested for the artifact-based RT relations and the capacities of the contribution format are by no means definitive. These could be tailored on a project-specific basis to account for existing RT practices and any current distinctions used in documentary practices. Use of the approach could further fit in with the work schedules of agents, since no constraints have been imposed on the order in which the information needs to be collected. Even if certain information is omitted, noting a solitary contributor would lead to some form of personnel-based RT, and later additions and changes could always be made without jeopardising the approach. However, the approach does not guarantee that only complete and accurate information will ever be recorded, though the information required could be subject to some checking as it is input to assist with this issue. With more experience in the application of the approach in practice, we could examine the types of exception most likely to occur when entering the information it requires and when requesting various forms of personnel-based RT, and so assess how these could best be accounted for by such checking.

### 7.5. Disruption to the Status Quo

The approach does not stop agents carrying out personnel-based RT in the current informal manner. If a subtle reason exists for this practice, perhaps because agents can learn all sorts of extra information by attempting to manually trace those who have been involved, this can still happen where needed. The approach just serves to guide this practice and, in making the contribution structure visible, also helps to legitimise the artifacts produced. Agendas which need to be hidden for political reasons can therefore remain so, whilst traceability back to contact points for such agendas would at least ensure their involvement in subsequent changes and the like. Whether or not the application of the approach would disrupt existing working practices in an
organisation would firstly depend upon whether RT was already implemented. It would also depend upon whether the basic tenet behind the approach complied with the prevailing organisational culture. Since the approach depends upon agents being open about their contributions, it could be undermined if accountability for one's contributions were used to attribute blame, rather than used to encourage process improvement. The range of information that could be inferred as a by-product of the approach could also be very sensitive in some organisational settings because, although the approach itself does not undermine organisational structures, subversions could be revealed through the information it provides. Furthermore, introducing the potential for personnel-based RT may upset existing practices, mainly because it would mean that agents remain active in a project for longer than they might have originally anticipated. To address such issues again suggests the need to put policies in place to govern the use of the approach and its results.

7.6. Accessibility from Mainstream Applications

Organisations do not require a specialised tool to apply the approach and use its results. Therefore, it need not obscure what are considered to be the mainstream activities in systems and software development. As most project artifacts are either textual, graphical, or compound in nature, we have suggested how the approach could be implemented as extensions to existing document preparation tools in conjunction with some form of underlying database system. With the introduction of open technologies which allow compound documents to be created using different applications, individual preferences would still be supported with regard to the applications used for producing artifacts. As the approach could also be implemented through simple extensions to document management systems or to contemporary RT tools, again it need not obscure the use of the other tools ordinarily used to carry out development activities. Furthermore, integration with those tools which support the communication likely to be required as a consequence of the personnel-based RT provided would, for example, enable tailored messages to be sent to those agents involved in the production of a requirement when proposing to change it.

7.7. User Involvement

The approach is grounded in an understanding of RT problems and was derived from a specification of requirements to alleviate these. These early phases of our development process involved many potential users. Although potential users were not directly involved when subsequently developing the approach, we periodically received comments, suggestions, and anecdotes from those that could be considered potential users. However, we were never in a position to examine, from first-hand experience, what actually happens when RT breaks down in projects. This means that we were unable to determine information like:
Whether the reason for needing to find the agents involved in a requirement's life influenced how they were identified and located.

What information about those involved ordinarily needed to be uncovered.

What generally took place once these agents had been identified and located.

Instead, tailoring the basic approach to a specific set of users and their working practices is now the subject of longer-term action research with practitioners from within a specific organisation.

7.8. Managing Uptake and Acceptance

We have identified some guidelines for managing the uptake of the approach and for encouraging its continued acceptance. For instance, as we have noted that existing development practices and organisational cultures are important factors that are likely to influence the uptake of the approach, this implies that the approach is more likely to be taken up in those organisations where an emphasis is increasingly being placed on RT to improve development processes or where project management concerns are turning towards the personnel involved in projects. It is also more likely to be taken up if it does not require specialised tools and training, so can be implemented as an addition to existing applications and working practices. Policies governing how the information is to be gathered, and how the information it provides is to be used, would be important to establish at the onset of a project to deal with user concerns and so encourage acceptance. Finally, acceptance would be more likely if the approach were first used to address the most pressing problems within particular projects and organisations, so some significant benefits could be immediately demonstrated, before extending its scope to cover more of what we have mentioned. Phasing in the approach would further enable those factors which are likely to promote and hinder its acceptance in each particular organisation or project to be uncovered and taken into account.

7.9. Evaluation and Generalisation

It has been difficult to evaluate the approach. As we have not quantitatively evaluated aspects like the usability of the proposed implementation and the usefulness of the personnel-based RT it provides, having obtained subjective opinions on such matters from a small set of practitioners, we cannot generalise from our findings. Each application of the approach is likely to have some dependence on the background and priorities of the individual organisations, projects, and agents involved. However, we have demonstrated the feasibility of the approach, in that it addresses a real problem in a practical way. We have also identified those factors which are crucial to its success and those factors which are most likely to cause it problems.
8. Summary of Part IV

In Part IV, we have described how we validated whether the approach based on modelling and using contribution structures provides a feasible and practical way to identify those who have participated in the RE process. We have described a prototype tool that was developed to implement the approach, having conducted an RE exercise for this tool in the process, and have further mentioned some basic considerations for a production version. We have demonstrated how this tool was then used to demonstrate and refine the approach, assisted by scenarios of use, and have outlined one such scenario to illustrate its operation and the added value it brings. We have also reported on a case study we conducted to examine details related to the application of the approach and the use of its results in actual practice. We have discussed the issues that arose and summarised the main implications we noted for practice. Based on all such experiences, we have provided a more general and critical evaluation of the approach.
Part V: Conclusions

1. Overview of Part V

In Part V, we present the conclusions of this thesis. We first re-examine the thesis argument that we presented in Part I and then go on to describe the contributions that we believe this thesis has made. We list a number of outstanding issues and point to how these could be addressed by either short-term or long-term research. We then outline our current agenda for the uptake of the approach in an industrial setting. We finish with a summary of Part V, followed by a brief summary of the entire thesis.

2. Thesis Argument Revisited

In Part I of this thesis, we laid out the structure of the thesis argument. We stated that the issue it was to tackle was that of how it would be possible to reduce the experience of RT problems in practice. In recognising that the so-called "RT problem" is multifaceted in nature, we stated that its position was that the greatest potential for reducing RT problems, especially in the longer-term, would be to address those more fundamental issues underlying extant problems. In Part II, we summarised the results of a problem analysis exercise that we carried out to identify the crux of the RT problem. In Part III, we described an approach that we developed to tackle this more central problem. In Part IV, we reported how we validated that the approach provides both a practical and feasible solution to this central problem. Although it follows from the thesis argument that the experience of RT problems in practice should be reduced as a consequence of using the approach, and although we have further indicated numerous other interrelated problem areas that should also be assisted, this is something that is difficult to assess conclusively without further long-term study. Such studies would need to compare the RT problems experienced in projects using the approach against control projects not using it. In addition, since the introduction of any proposed solution is likely to impact the nature of existing RT-related tasks, such studies would also need to re-examine the knock-on effects when using the approach with regard to the underlying nature of the RT problem.
3. Contributions

We believe this thesis has made a direct contribution to the field of RE via the numerous contributions it has made to the particular area of RT. In turn, we believe it has made a more general contribution to the discipline of software engineering and, by extension, to the discipline of systems engineering. In this section, we first explain how this contribution to software engineering has been made, and then go on to describe each of the more specific contributions to RT, which we first outline below:

1. A review of the state-of-the-art in RT, including a critique of current support, plus a description of contemporary research.

2. An empirically grounded analysis of what the so-called "RT problem" actually is.

3. The differentiation of two fundamental types of RT, pre-RT and post-RT, along with an explanation as to why there is a need to focus more research effort on the issues related to pre-RT.

4. Uncovering the shortcomings of conventional artifact-based RT and revealing the crux of the RT problem to be identifying the human sources of requirements and requirements-related information.

5. The development of an approach to tackle the crux of the RT problem, based on modelling and using the contribution structure underlying requirements artifacts, which makes personnel-based RT possible.

6. Demonstrating the wider impact of the approach with regard to issues like quality development and process improvement.

3.1. To the Discipline of Software Engineering

To explain how this thesis has made a contribution to the discipline of software engineering, we make reference to the definitions of software engineering and its sub-fields, given in Boxes 32 and 33 respectively.
1. **Software engineering** is that form of engineering that applies:
   - a systematic, disciplined, quantified approach,
   - the principles of computer science, design, engineering, management, mathematics, psychology, sociology, and other disciplines as necessary,
   - and sometimes just plain invention,
   to creating, developing, operating, and maintaining cost-effective, reliably correct, high-quality solutions to software problems.

2. **Software engineering** is also the study of and search for approaches for carrying out the activities of (1) above."

**Box 32:** A definition of the term "software engineering" (taken from [Berry 1992]).

"1. **Theory of programs and programming** - ...

2. **Formal methods** - ...

3. **Technology** - the discovery, development, and validation of the effectiveness of software tools to help carry out one or more steps of the programming process, including the application of formal methods, for the purpose of improving the ability of the user to produce quality software.

4. **Methodology** - the discovery, development, and validation of the effectiveness of nonformal but systematic manual procedures for the purpose of increasing the ability of the applier of the procedures to produce quality software.

5. **Management** - the discovery, development, and validation of the effectiveness of managerial techniques to help people and groups of people produce quality software.

6. **Production of software artifacts** - ...

**Box 33:** Definitions of the sub-fields of software engineering in which we have contributed (taken from [Berry 1992]).

This thesis has contributed to the discipline of software engineering as it has proposed and demonstrated an approach, which draws together principles from a number of relevant disciplines, to address a problem which we have shown to have a critical impact on the production and maintenance of quality software. More specifically, it has contributed in terms of:
(1) **Methodology**, as it has provided an approach which can be applied to increase the likelihood of quality software being created and maintained.

(2) **Management**, as it has provided an approach which can be used to assist the social aspects involved in creating and maintaining quality software.

(3) **Technology**, as the approach has been shown to be highly amenable to tool support.

### 3.2. Surveying the State-of-the-Art in RT Research and Practice

We have organised our review material in a practically informed way to clarify the current state of RT practice. We have distinguished between the different forms of RT support, provided a critique of their scope, strengths, and weaknesses, and have further discussed the main directions of contemporary research. We believe this review provides a currently lacking overview of the state-of-the-art in RT.

### 3.3. Understanding RT and the So-Called "RT Problem"

We have carried out an analysis of the RT problem and have thereby advanced some understanding of what actually lies behind the ubiquitous and continued cries of an RT problem by practitioners. Through so doing, we have provided a working definition of RT, which has since been adopted by others, like [Loucopoulos & Karakostas 1995, Ramesh et al. 1995]. We have also highlighted two fundamental types of RT, namely pre-RT and post-RT, that were not previously distinguished. These types have since been appealed to by others, like [Laubengayer & Spearman 1994, Leite & Oliveira 1995], and support for artifact-based pre-RT is gradually being introduced in the leading commercial tools. We have also shown how pre-RT and post-RT together provide a conceptual framework through which to understand the many issues found to underlie the so-called "RT problem". Using this framework, we have been able to examine where the most leverage for RT improvement lies, as well as to identify the cause of longer-term RT problems.

### 3.4. Uncovering the Need to Focus Effort on the Issues of Pre-RT

We have shown how conventional notions of RT have, to date, been restricted to deal with the concerns of post-RT. Moreover, we have explained why advances in post-RT will only bring about restricted improvements in RT, and so argued that research efforts need to be re-focused to deal with the issues relating to pre-RT if we are to provide a comprehensive approach to RT. In particular, we have indicated why this re-orientation is essential if we are to establish firmer foundations for quality-driven development and maintenance. We have further identified the critical problem areas confronting pre-RT improvements, outlined the key requirements to address
these issues, and have pointed to potential solution options and research ideas for many aspects. Pre-RT has since been acknowledged elsewhere as the critical issue facing RT improvement [Kavakli 1995].

3.5. Identifying the Crux of the RT Problem

Despite the improvements that we suggest can be realised by focusing more effort on addressing the issues of pre-RT, we have highlighted the eventual limitations of a purely artifact-based form of RT, even if it deals with numerous relations between the most extensive types of project artifact at very fine levels of granularity. This is because we have pointed to the inevitability of those situations in which practitioners will need to augment any information retrieved through RT schemes with information obtained through additional forms of communication, be this information about requirements deployment or, as we have found more crucially, about requirements production and refinement. We have further indicated that the ability to identify the appropriate human sources with whom to conduct this communication is central to the longer-term RT problems. We have explained how this is, not only a fundamental and problematic working practice, but a working practice which the prevailing approaches to RT and the thrust of current RT research strives to eliminate. We have thereby demonstrated a crucial need to tie people into this predominantly artifact-based RT equation.

3.6. Introducing Contribution Structures and Personnel-Based RT

We have introduced the novel concept of the contribution structure to anchor requirements artifacts in the social structure that gave rise to them. We believe that contribution structures provide the strongest and most fundamental anchor for RT improvement because we have found that people are often the final authority about requirements and, as such, are frequently able to prevent claims of RT problems. We have described a generative model that can be used to handle the dynamic and emergent nature of these contribution structures and have described an approach to define, maintain, and make use of this model, which we have further shown to be both practical and feasible. In turn, we have shown how this enables conventional forms of artifact-based RT to be extended with personnel-based RT, thus providing a practice to fall back on when either: (a) economy of documented RT information is paramount; (b) RT information is absent; or (c) RT fails. We have further demonstrated how personnel-based RT makes it possible to selectively and rapidly identify the most appropriate agents to involve and inform in situations of change, decision making, requirements query, quality assessment, conflict resolution, and so forth.

3.7. Providing Additional and Far-Reaching Consequences

We believe this thesis also makes a contribution in that it opens up the potential for much follow-on research. This could range from pursuing the notion of an RT markup language through to
examining the analytical potential to be gained from providing contributor and contribution profiles of projects. Furthermore, we believe the approach has a more general applicability, as it could be incorporated as a component of collaborative writing tools and the like. Below, we explain how we believe the approach provides: (a) the firmest of foundations for quality-driven systems and software engineering; (b) firmer foundations for many contemporary forms of project analyses; and (c) the foundations through which to inform RE process improvement.

3.7.1. Firmest Foundations for Quality Development

Quality-oriented approaches to software development, and their increasing support by CASE tools, have become the focus of considerable attention [Jarke & Pohl 1992]. Although they have led to marked improvements in software quality, as reported in [Aaen et al. 1992, Polack 1990, QED 1989], this has rarely been to the extent anticipated. Some reasons for this are discussed in [Curtis 1992, Sumner 1992, Wynekoop 1992]. In this thesis, we have argued that RT is a major quality-enabler. We have explained the restricted impact that post-RT can have on quality and have explained why pre-RT extensions are required to provide a firmer foundation upon which to achieve and assess quality. Together, they enable the concept of "quality", and similarly its satisfaction criteria, to evolve with the evolving environment in which "quality" is defined and measured, and the evolving development process through which it takes effect. We have also explained how traceability of the personnel who have contributed to the requirements produced in the RE process supports a fundamental working practice which is often the only way to explain and assess quality criteria. Therefore, by further anchoring requirements to their contribution structure, we provide the firmest of foundations upon which quality systems and software can be built and measured.

3.7.2. Firmer Foundations for Contemporary Analyses

By anchoring requirements to their contribution structure, we also offer a more fundamental basis for carrying out various forms of organisational-based modelling and analysis, as it has frequently been said that organisations can only be really understood by locating them in their social context [Morgan 1990]. As a particular example, we believe the approach provides a better basis for the many speech act forms of analyses, these being based on the work of [Austin 1962, Searle 1969]. Such analyses are often carried out to examine, record, and classify the communication that has taken place in the development process, as typified by the Coordinator tool [Winograd 1987/8, Winograd & Flores 1986]. The information provided through the use of our approach provides a firmer basis for such analyses because it has been observed that knowledge of the underlying social network is a prerequisite for speech act analysis [Ervin-Tripp 1973, Fishman 1972]. This is because agents communicate as the incumbents of social roles, which obviously impacts the illocutionary force. So, to distinguish a "representative" speech act as one of "asserting" for the first time from one of "summarising", details about both the informational and social arrangements
need to be known \textit{a priori}. Furthermore, the ability to identify implicit and derived group contributions, in addition to the explicit ones, means that more suitable forms of group-based analyses can be invoked under appropriate circumstances [Hughes 1984].

3.7.3. Foundations for RE Process Improvement

We need a dependable process if we are to deliver dependable systems and software. This of course requires a better understanding of the actual process. The approach provides a research vehicle through which we can learn more about the actualities of RE processes and working arrangements, and so uncover details relating to best practice. With longitudinal studies of use of the approach, including project post-mortems and cross-comparisons, we would be in a better position to examine issues like:

1. What happens to and with requirements artifacts in practice. We could use such information to guide the development of future requirements methods, models, and languages.

2. How useful many of the widely-used concepts which typically guide a project's management really are, like the static organisational models, the predefined role structures, and the imposed work breakdown structures. By linking contribution structures to such models, we could compare expectations with reality, and so inform future such modelling activities.

3. How the contribution structure behind a requirement grows, changes, and emerges over time. We could use such information to assess the impact that different working arrangements are likely to have on the quality of the end product, and to further uncover those that either compound or alleviate RT problems.

4. How we can reuse requirements. This is because our empirical studies noted that reuse of requirements is more likely to occur in a project when those originally responsible for the requirement in question, as well as those responsible for any related artifacts being reused as a consequence, are either the ones reusing it or are at least directly accessible to them.

4. Outstanding Issues and Future Work

In this section, we describe the outstanding issues with regard to our approach to address the crux of the RT problem, and then indicate how they could be addressed by further work. This is first summarised in Table 37. Note that these issues mainly stem from the assumptions we have made, and the boundaries we have drawn, to contain the problem and demonstrate the concepts we have proposed.
Part V: Conclusions

<table>
<thead>
<tr>
<th>Incomplete coverage of the RE space and RE participants by the approach.</th>
<th>Use of predefined and non-empirically informed categories in the approach.</th>
<th>The organisational resistance facing any uptake of the approach.</th>
<th>Many wider considerations related to approach implementation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accounting for indirect contributors to artifacts and artifact recipients.</td>
<td>Accounting for those contribution capacities and social roles used in practice.</td>
<td>Encouraging a change in organisational culture, potentially through policies.</td>
<td>Dealing with aspects like the physical and conceptual distribution of artifacts and agents.</td>
</tr>
<tr>
<td>Accounting for evolving requirements artifacts, versions, and configurations.</td>
<td>Accounting for those artifacts and artifact-based RT relations used in practice.</td>
<td>Automatically capturing as much information for the approach as possible.</td>
<td>Improving RT visualisation and navigation through project spaces.</td>
</tr>
<tr>
<td>Accounting for requirements discussions to guide this process.</td>
<td>Accounting for types of requirements change and their impact.</td>
<td>Instantiating the approach in contemporary project tools and practices.</td>
<td>Determining the generalisability and scaleability of the approach and results.</td>
</tr>
</tbody>
</table>

**Table 37: Summary of outstanding issues and future work.**

4.1. Incomplete Coverage

The coverage of the approach has been restricted for the purpose of this thesis. It only focuses on the tangible artifacts produced in RE because the purpose of this thesis has not been to investigate ways in which we can provide a total recording of what has gone on in the RE process. We maintain that this is not realistic, short of conducting and recording the entire RE process within a virtual environment which can be replayed at leisure. The approach only focuses on those agents who have been directly involved in producing these requirements artifacts. In developing the approach further, it would be necessary to also account for those agents who have been indirect participants in the production of requirements artifacts, as well as those agents to whom the requirements artifacts have been directed. Issues remain with regard to handling the evolution of requirements artifacts and their subsequent impact on the contribution structure. Issues also remain with regard to providing guidance or control in the application of the approach.

4.1.1. Accounting for Indirect RE Contributors and Recipients

Although a boundary needed to be drawn to restrict a potentially vast social structure, this was specifically chosen to focus on the participant production roles of Goffman's interactional framework repeated in Table 38. Future work would be to extend this boundary to encompass the remaining aspects. To account for the non-participant production roles, the approach could be extended to capture distinctions like the ultimate source of authority, the other users for whom a particular user has devised a requirement, and so forth. However, such indirect contributors are not as immediately apparent, may be difficult to exhaustively determine, and could be huge in...
number. We could account for the reception roles in a similar way to the production roles if we recall that Goffman also decomposed the notion of "hearer" to delineate those who are ratified and those who are not ratified [Goffman 1979]. In undertaking to account for the full framework, we would certainly enrich the approach, and would further be in a position to examine the changes in alignment of all those involved throughout the course of a requirement's life.

<table>
<thead>
<tr>
<th>Participant roles</th>
<th>Production roles</th>
<th>Reception roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Those agents directly involved in producing RE artifacts.</td>
<td>Those agents who make use of RE artifacts for whom they have been explicitly produced.</td>
<td></td>
</tr>
<tr>
<td>Non-participant roles</td>
<td>Those agents indirectly involved in producing RE artifacts.</td>
<td>Those agents who make use of RE artifacts for whom they have not been explicitly produced.</td>
</tr>
</tbody>
</table>

Table 38: Partitioning the social dimension of RE (a repetition of Table 9 from Part III).

4.1.2. Accounting for Evolving Requirements

Boundaries also needed to be drawn to reduce the complexity of the artifact space we were dealing with so we could focus on demonstrating the basic concepts of the approach. For this reason, we have only been concerned with the initial production of the tangible requirements artifacts, as opposed to their adoption and evolution. Although our approach can handle versions using the adopts artifact-based RT relations, there are likely to be other factors to consider which have implications for the ability to maintain the integrity of the model of the contribution structure over time, like providing more practical ways to account for the differences between the original production, subsequent use, and possible revision of these artifacts. Version control and configuration management techniques obviously need to be integrated with the approach. These aspects have been the focus of plenty of work in RT [Macfarlane & Reilly 1995, SSSL 1995a].

4.1.3. Accounting for Discussion About Requirements

It would be a relatively straightforward exercise to extend our approach with schemes which support discussion about requirements artifacts, be this for requirements elaboration purposes [Butcher 1988, Finkelstein & Fuks 1989], or for requirements revision purposes [Finkelstein 1992, Golipour-Koujali 1992]. This is because such activities could be conceptualised as dialectical discussions, with commitment stores indicating the current state of play in the dialogue at each stage, and legal moves guiding the evolving dialogue structure [Hamblin 1971, Mackenzie 1981]. From the use of our approach, commitment stores could be determined and maintained to record an agent's commitments to both artifacts and to other agents throughout the RE process. Based on a better understanding of the various moves that take place in the elaboration and revision of the requirements artifacts, we could provide rules to inform how these commitments can be made,
refined, altered, and lead to others, and so support and guide the RE process.

4.2. Preliminary Categories

The categories we have chosen to delineate the types of contribution capacity, artifact-based RT relations, and social roles, are only preliminary and have been chosen to demonstrate the basic concepts of the approach. Further work would be to uncover those categories and distinctions that practitioners actually use and appeal to in practice. To acquire such categories, in-depth field studies of actual working practices and arrangements would be essential. These could be conducted using ethnographic or ethnomethodological studies [Button 1991, Garfinkel 1967, Hammersley & Atkinson 1983], and by using techniques like participant observation [Friedrichs & Lüdtke 1975], conversation analysis [Goodwin & Heritage 1990, Sacks 1995], interaction analysis [Kendon 1990], and discourse analysis [Gumperz 1982, Stubbs 1983]. With a better understanding of RE work, we would be in a position to select those categories which would be the most useful to practitioners and thereby cause fewer conceptual problems.

4.2.1. Accounting for Actual Contribution Relations and Social Roles

Problems may arise from the reliance on people to instantiate the capacities of the contribution format, as well as from their ability to characterise the contributions according to our scheme. As mentioned above, it would be useful to uncover any contribution capacities that the RE participants themselves orient to whilst in the process of producing requirements artifacts, and so use these in the contribution format instead. Similarly, it would be useful to appeal to any social roles, role relations, and commitments that are found to be prevalent in the domain of RE. It would also be interesting to examine how others' suggestions for such categories could be instantiated in the approach, like Benne and Sheats' comprehensive list of the social roles considered necessary for carrying out and sustaining group work [Benne & Sheats 1948]. In addition, further work could be to examine how different degrees of contribution could be identified and incorporated into the approach. Even a crude distinction between strong or weak contributions could provide information to inform about priority access structures for agents, critical or optional role relations between agents, and so open up more options for personnel-based RT.

4.2.2. Accounting for Actual Artifacts and Artifact-Based Relations

Problems may also arise when attempting to identify the finer-grained types of artifact-based RT relation. Each participant is likely to have a different interpretation of such relations because the communicative function of language will be related to the purposes and needs of each participant [Saville-Troike 1982]. This multiplicity should be accounted for in the approach and would need to be examined further. As mentioned above, it would be useful to uncover the basic units which
practitioners use to organise knowledge in the RE domain, as well as to uncover details as to how these units decompose and interact to form larger structures. Such information could be used to inform a suitable language for talking about requirements artifacts, their different functions, and their various types of interrelation. In addition, it would be interesting to examine how knowledge of the primary and subsidiary nature of these requirements artifacts, like an artifact and its rationale, could be incorporated into the approach to exploit this knowledge further. Another point is that we have not been concerned with recording details about those agents that put the various artifact-based RT relations in place. Again, this would be an interesting aspect to examine further and to take into account in the approach.

**4.2.3. Accounting for Types of Requirements Change**

Keeping the contribution structure of a requirements artifact up-to-date is complicated by the fact that the nature of any change to the artifact concerned will have implications on how this recomputation should be done. Future work would be to examine how our approach could be extended to account for this issue. Here, we could examine existing taxonomies which claim to delineate the types of change that can occur to requirements, such as that given in [Harker *et al.* 1993, Rolland 1994a], to see how different types of change would impact the underlying contribution structure. Alternatively, if we incorporated the artifacts which both instigate and result from a change as part of the approach, we would be in a position to carry out retrospective studies to learn about the types of change which actually occur in the RE process, along with their effects. This information could be fed back into improving the requirements change control process.

**4.3. Organisational Resistance**

A further outstanding issue is that of organisational resistance. As we have previously mentioned, there is likely to be some need for changes in organisational and project culture if the approach is to be accommodated in practice. We have also indicated that the approach is likely to be more acceptable if suitable policies are put in place to de-politicise the issues, if ways are found to automatically capture as much of the information the approach requires as is possible, and if the approach can be applied from within commonly used project tools and working practices.

**4.3.1. Changing Organisational Policies**

Many organisations have yet to implement a culture for artifact-based RT, let alone for personnel-based RT. We therefore need to examine, not only how this change in culture can be encouraged, but also how it can incorporate concern for our extensions. An incentive is surely the direct benefits the approach brings by way of personnel-based RT, as well as the further forms of analysis it makes possible. Even with incentives, the approach would need to be introduced in an
incremental manner, rather than in its fully-fledged form. This is so that any immediate changes to working practice can be minimised, so that suitable policies can be established and tested, and so that the benefits reaped with each increment can be demonstrated. The problems faced here are not unlike those faced by today's project management techniques and tools. It has been reported that, although many of today's companies invest in project management systems, reward from such investments will not come about until accompanied by a company-wide culture for project management which has put workable practices and procedures in place [Archibald 1995].

4.3.2. Automatically Capturing the Information Needed

Whose responsibility should it be to record the contribution format of an artifact? Should it be the responsibility of those involved in producing the artifact or should it be the job of a dedicated agent? When exactly should this information be captured? Should it be done during artifact production or once it has been completed? However these issues are tackled, problems are likely to arise from the reliance on people to establish the original relations between agents and artifacts, even though they can be subsequently managed and used to infer others. One reason is because, unless captured unobtrusively and as a by-product of other working practices, a post hoc rationalisation is likely to introduce scope for error. Therefore, we either need to examine how we could provide the necessary conditions to promote the manual exchange of accurate information at the time of working, else examine ways in which more of these details could be captured automatically. For instance: (a) the principal could be determined from signatures on artifacts; (b) the documentor could be determined from those physically constructing artifacts in an on-line and password-secure environment; and perhaps (c) the author could be determined as a consequence of some form of agent modelling based on heuristics found in authenticity studies to recognise familiar author styles. It might also be possible to monitor for linguistic cues and paralinguistic markers in the artifacts themselves to flag shifting contribution formats. Future possibilities for automatically identifying the semantics of the artifact-based RT relations may also come from work investigating how coherence and cohesion relations can be used to computationally generate and organise text [Mann 1984, McKeown 1985].

4.3.3. Instantiation in and Integration with Third-Party Tools

With minimal extra work, the approach could be incorporated in existing tool support for RE and RT as these are likely to contain some form of document preparation system and database technology through which the approach could be instantiated. For instance, in dedicated RT tools like DOORS, this could be done by choosing suitable attributes and writing the necessary DXL scripts. In end-to-end development tools like Cradle, this could be done by providing suitable category codes and item fields. Also, as more and more organisations appear to be developing document management strategies through which their information can be codified, indexed, and made amenable to sophisticated forms of analysis [B & T 1995, CGSA 1995a], and as this trend...
appears to be accompanied by a growing use of supportive products and services [CGSA 1995b, Cimtech 1995, Seachrist 1995], the approach could be incorporated as part of such in-house strategies. As the return on investment from document management largely depends on the indexing schemes used, we suggest that contributors would be a highly valuable indexing scheme. A future exercise would also be to examine how suitable communication tools could be coupled with the approach to support the activities required as a consequence of the personnel-based RT. For instance, by storing preferred communication protocols with each agent's details, and through integration with platforms like that described in [Cockburn & Greenberg 1993], the approach could offer the potential to automatically instigate any required communication with identified human sources across the most appropriate communication channels.

4.4. Implementation Issues

There are a number of outstanding issues related to the implementation of the approach that would need to be looked into in future work. For instance, how could the approach account for the physical or conceptual distribution of requirements artifacts and agents? Also, what could be done to improve the visualisation and navigation aspects of the personnel-based RT? In addition, as we have proved neither the scaleability nor generalisability of the approach, these remain aspects that would need to be explored further.

4.4.1. Dealing with Physical and Conceptual Distribution

Although the provision of a shared, consistent, and coherent scheme for all to use could imply that the approach provides support for distributed working, we have not investigated the full technical ramifications of implementing our approach in a distributed environment or with a distributed project repository. Here, it would be essential to determine how artifact-based RT itself could be established and maintained in a distributed setting, as this is a subject which has not been examined to date in the detail it deserves.

4.4.2. Improving RT Visualisation and Navigation

Many issues arise related to the most appropriate way in which to present the results of artifact-based RT and personnel-based RT. Here, more experimentation would need to be carried out with the potential end-users to explore suitable visualisation techniques to differentiate between individual and group contributions, to depict the various types of traceability made possible, and so forth. Issues also arise as to the most suitable techniques and metaphors to use to navigate the project space and its associated contribution structure. Here, the use of spatial metaphors and virtual reality come to mind, so providing a way to immerse end-users in the requirements information itself so that they can physically move through it to achieve RT.
4.4.3. Determining Scaleability and Generalisability

Although we have attempted to demonstrate scaleability using modelling techniques, there could be some unforeseen issues related to the size and longevity of projects that affects the feasibility of the approach. Other unforeseen issues could surface from generalising the approach to encompass the entire systems and software lifecycle and so from modelling the contribution structure in all phases of development and maintenance. These issues could only be uncovered via more case studies with projects having such characteristics. Case studies would also be essential to examine issues like: (a) at what point the benefits of the approach accrue; (b) how much time investment is really needed; (c) any thresholds; and perhaps most importantly (d) whether the potential for informal communication with appropriate agents statistically reduces RT problems, speeds up the change process, improves quality, and so forth.

5. Technology Transfer

Alford, in [Alford 1994], maintains that successful technical transfer depends on:

(1) Not having too complex an idea.

(2) Being able to show immediate beneficial consequences of this idea.

(3) Being able to try a little bit of it at a time and gradually extending its scope.

(4) The realisation of the idea being perceived as an extension to known technology, so harnessing existing skills rather than requiring completely new ones.

We believe that the underlying idea of the approach is fundamentally very simple and have pointed to how the complexity of the scheme and the forms of analysis that it makes possible can be extended incrementally and as desired. We have also shown the beneficial consequences of the approach, not only directly and to those involved, but also in the form of process improvement. Finally, we have explained how, by using existing tools, the approach could be physically put into operation with minimal extensions to existing RT practices.

This said, we finally point out that the approach has recently captured the interest of industry, and that a project has just been started up between academia and industry to effect such technology transfer. The industrial partner view RT as a key problem area and see the approach as a potential means through which many of their particular RT problems could be addressed. The research will therefore aim to investigate how the approach can actually be transferred to an industrial setting and will then make the necessary refinements for it to be realised in practice. This project will provide the opportunity to examine the outstanding issues mentioned above, to conduct more experimentation, and to perform the further long-term industrial case studies and evaluation that
Part V: Conclusions

6. Summary of Part V

In Part V, we have re-examined the argument of this thesis, as well as assessed its merit. We have also described the various contributions we believe this thesis has made, to the areas of RT and RE in particular, as well as to the more general disciplines of systems and software engineering. We have listed a number of outstanding issues with the approach and its validation and have outlined some of the research issues that have been revealed as a consequence. We have mentioned how the approach could be taken up by practitioners and finally indicated that such technology transfer is in fact in progress.

7. Thesis Summary

In Part I of this thesis, we presented a statement of the thesis argument and introduced the topic of RT. We situated RT within the discipline of systems and software engineering and demonstrated how it is integral to RE. We indicated why RT is crucial in the drive for quality systems and software, but pointed to evidence of an RT problem. We explained both how and why there is a problem with current RT practice and a shortcoming with contemporary research directions. We went on to argue that the relatively slow progress we have seen to date with the RT problem is due to a poor understanding about what the RT problem actually is. We suggested a thorough analysis of the RT problem as the best way forward.

In Part II of this thesis, we analysed the multifaceted nature of the so-called "RT problem" that many practitioners refer to, and went on to explain why there is unlikely to be an all-encompassing solution. We outlined the method we used for collecting data about the problem and, from this data, we showed how the problem itself can be decomposed into a number of more fundamental underlying problems. We further indicated how these problems stem from conflicting and changing viewpoints regarding what RT is, what the problems with it are, and what it is meant to achieve. Through the provision of a conceptual RT framework, we introduced a distinction between pre-RT and post-RT, and described their current support and potential for alleviating RT problems. We argued that, in order to achieve any order of magnitude improvement with RT problems in the longer-term, there is a need to re-focus research efforts on addressing the issues of pre-RT. With this objective, we discussed the problems confronting, and the requirements for attaining, such advances in pre-RT. We then indicated how some of these requirements could already be met and made suggestions for additional progress. Although the obvious direction appeared to be towards gathering extensive requirements-related information and making this traceable, we argued that this needs to be augmented with the ability to identify those individuals and groups in a position to supplement it. This is because we found that the ability to trace those who have contributed in RE is a fundamental working practice which provides what is often the
only way to explain and assess change, reason about real needs, deal with the inevitable shortcomings of information which is documented, and so abate RT problems. The inability or difficulty in identifying the human sources of requirements, requirements-related information, and requirements-related work was found to be the crux of longer-term RT problems. We suggested the need to tie the appropriate people into the RT equation to address this central problem.

In Part III of this thesis, we provided a detailed description of an approach to address the crux of the RT problem. We described how it makes relevant details about the social structure that gave rise to the tangible artifacts produced in RE explicit and so makes personnel-based RT a possibility. We outlined some frequently asked questions about RE participants that are problematic to answer and described the current practice which leads to such difficulties. We then listed the fundamental requirements for an approach to tackle this problem, though highlighted some political issues that need to be considered. We detailed each of the steps of the approach, summarised the foundations upon which they are based, and illustrated their operation through examples. In particular, we demonstrated how the approach is able to build an increasingly detailed picture of the contribution structure from minimal extra input, and to the different levels of detail required. In so doing, it makes a useful portion of the social structure that gave rise to requirements artifacts explicit and so traceable. We further provided a model-based specification of the approach to clarify how contribution structures can be modelled and used. Based on this, we demonstrated how the approach can be used to address the frequently asked questions, as well as pointed out the opportunities it presents for additional forms of analysis.

In Part IV of this thesis, we described how we validated whether the approach provides a feasible and practical solution to the crux of the RT problem. We described the prototype tool that we developed to implement the approach, having conducted an RE exercise to do so, and further mentioned some basic considerations for a production version. We then described how the prototype tool was used to demonstrate and refine the approach with scenarios of use. We outlined one such scenario to illustrate its operation. We then reported on a case study we conducted to examine details related to the application of the approach and the use of its results in practice. Here, we discussed the issues that arose and the implications for practice. Based on all such experiences, we provided a critical evaluation of the approach.

In Part V of this thesis, we re-examined the thesis argument. We described the various contributions this thesis has made, to RT and RE in particular, and to systems and software engineering in general. We listed a number of outstanding issues with the approach, its implementation, and its validation. For each of these issues, we went on to outline further research. We finished by mentioning our current technology transfer agenda.
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALBERT</td>
<td>Agent-Oriented Language for Building and Eliciting Requirements for Real-Time Systems.</td>
</tr>
<tr>
<td>ARTS</td>
<td>Automated Requirements Traceability System.</td>
</tr>
<tr>
<td>CASE</td>
<td>Computer-Aided Software Engineering.</td>
</tr>
<tr>
<td>DIF</td>
<td>Document Integration Facility.</td>
</tr>
<tr>
<td>DRL</td>
<td>Decision Representation Language.</td>
</tr>
<tr>
<td>DXL</td>
<td>DOORS Extension Language.</td>
</tr>
<tr>
<td>DOORS</td>
<td>Dynamic Object-Oriented Requirements System.</td>
</tr>
<tr>
<td>gIBIS</td>
<td>Graphical Issue-Based Information System.</td>
</tr>
<tr>
<td>HTML</td>
<td>Hypertext Markup Language.</td>
</tr>
<tr>
<td>IBIS</td>
<td>Issue-Based Information System.</td>
</tr>
<tr>
<td>IEF</td>
<td>Information Engineering Facility.</td>
</tr>
<tr>
<td>IEM</td>
<td>Information Engineering Methodology.</td>
</tr>
<tr>
<td>IORL</td>
<td>Input/Output Requirements Language.</td>
</tr>
<tr>
<td>ISHYS</td>
<td>Intelligent Software Hypertext System.</td>
</tr>
<tr>
<td>ODA</td>
<td>Office Document Architecture.</td>
</tr>
<tr>
<td>ORDIT</td>
<td>Organisational Requirements Definition for Information Technology.</td>
</tr>
<tr>
<td>PDM</td>
<td>Planning and Design Methodology.</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>Post-RT</td>
<td>Post-Requirements Traceability.</td>
</tr>
<tr>
<td>Pre-RT</td>
<td>Pre-Requirements Traceability.</td>
</tr>
<tr>
<td>PRO-ART</td>
<td>Process-Based Approach to Requirements Traceability.</td>
</tr>
<tr>
<td>QFD</td>
<td>Quality Function Deployment.</td>
</tr>
<tr>
<td>QIP</td>
<td>Quality Improvement Paradigm.</td>
</tr>
<tr>
<td>RAD</td>
<td>Rapid Application Development.</td>
</tr>
<tr>
<td>RDD</td>
<td>Requirements-Driven Design System.</td>
</tr>
<tr>
<td>RE</td>
<td>Requirements Engineering.</td>
</tr>
<tr>
<td>REMAP</td>
<td>Representation and Maintenance of Process Knowledge.</td>
</tr>
<tr>
<td>RESPECT</td>
<td>Requirements Specification and Traceability Methodology.</td>
</tr>
<tr>
<td>RML</td>
<td>Requirements Modelling Language.</td>
</tr>
<tr>
<td>RS</td>
<td>Requirements Specification.</td>
</tr>
<tr>
<td>RSL</td>
<td>Requirements Statement Language.</td>
</tr>
<tr>
<td>RT</td>
<td>Requirements Traceability.</td>
</tr>
<tr>
<td>RTM</td>
<td>Requirements Traceability and Management System.</td>
</tr>
<tr>
<td>RTML</td>
<td>Requirements Traceability Markup Language.</td>
</tr>
<tr>
<td>SGML</td>
<td>Standard Generalized Markup Language.</td>
</tr>
<tr>
<td>SLATE</td>
<td>System Level Automation Tool for Engineers.</td>
</tr>
<tr>
<td>SODOS</td>
<td>Software Documentation Support.</td>
</tr>
<tr>
<td>StP</td>
<td>Software Through Pictures.</td>
</tr>
<tr>
<td>TAGS</td>
<td>Technology for the Automated Generation of Systems.</td>
</tr>
</tbody>
</table>
References

[Aaen et al. 1992]

[ALC 1994]

[Alford 1977]

[Alford 1993]

[Alford 1994a]

[Alford 1994b]
[Allen 1983]

[Allen 1984]

[Anderson 1989]

[Archibald 1995]

[Auramaki et al. 1988]

[Austin 1962]

[Avouris & Gasser 1992]

[Badini & Whitehouse 1989]

[Bailin et al. 1990]

[Balzer 1985]

[Balzer et al. 1983]
[Banton 1965]  

[Barnes & Edge 1982]  

[Barthes 1979]  

[Bateson 1972]  

[Bell et al. 1977]  

[Bench-Capon et al. 1991]  

[Benne & Sheats 1948]  

[Berners-Lee & Connolly 1993]  

[Berry 1992]  

[Bersoff & Davis 1991]  
References

[Biddle & Thomas 1966]

[Bijker et al. 1987]

[Biriotti & Miller 1993]

[Blyth et al. 1993]

[Bocker & Herczeg, 1990]

[Bødker 1989]

[Boehm 1976]

[Boehm 1981]

[Boehm 1986]

[Boehm et al. 1978]

[Bolter 1991]
References

[Bond 1990]

[Bond & Gasser 1988]

[Bonnycastle 1991]

[Borstler & Janning 1991]

[Bowen et al. 1990]

[Brady & DeMarco 1994]

[Brooks 1982]

[Brown et al. 1992]

[Brown 1991]

[BSI 1987]

[B & T 1995]
[Bubenko et al. 1994]

[Buckley & Poston 1984]

[Burke 1969]

[Burns 1993]

[Butcher, 1988]

[Button 1991]

[Buttny 1993]

[CACM 1992]

[Callon et al. 1986]

[Castelfranchi 1993]

[CGSA 1995a]
[CGSA 1995b]

[Chartier 1992]

[Chikofsky & Rubenstein 1988]

[Christophides et al. 1994]

[Chudge & Fulton 1994]

[Cimtech 1995]

[CMS 1993]

[Cockburn & Greenberg 1993]

[Conklin & Begeman 1988]
References

[Cooke & Stone 1991]

[Crane 1972]

[Crystal 1987]

[CT 1992]

[Culler 1981]

[Curran et al. 1994]

[Curtis 1992]

[Curtis et al. 1988]

[Curtis et al. 1992]

[Dardenne et al. 1993]
[Davis 1990]

[Davis 1994]

[Davis & Freeman 1991]

[De Beaugrande & Dressler 1981]

[Dell 1986]

[De Marco & Lister 1987]

[Dewire 1994]

[Diaper 1989]

[Dobson & Strens 1994]
[DoD 1988a]  

[DoD 1988b]  

[Dorfman 1990]  

[Dorfman & Flynn 1984]  

[Dorfman & Thayer 1990]  

[Dowson 1987]  

[Drew & Wootton 1988]  

[Dubois 1990]  

[Dubois 1994]  
[Duke & Harrison 1995]

[Easterbrook 1991]

[Edwards & Howell 1991]

[Ehn 1988]

[Ellis 1979]

[Ellis & Nutt 1980]

[Emmet 1966]

[Ervin-Tripp 1973]

[ESA 1987]

[Evans 1989]
[Feather 1991]

[Fickas & Finkelstein 1993]

[Fikes 1982]

[Finkelstein 1991a]

[Finkelstein 1991b]

[Finkelstein 1992]

[Finkelstein & Fuks 1988]

[Finkelstein et al. 1987]

[Finkelstein et al. 1990]
[Finkelstein et al. 1992]

[Fischer 1991]

[Fisher 1991]

[Fishman 1972]

[Flatten 1992]

[Flynn 1992]

[Flynn & Dorfman 1990]

[Foucault 1979]

[Fraser et al. 1991]
[Friedrichs & Lüdtke 1975]

[Fuks et al. 1991]

[Garfinkel 1967]

[Garg & Jazayeri 1994]

[Garg & Scacchi 1989]

[Garg & Scacchi 1990]

[Gasser 1986]

[Gasser 1991]

[Gasser & Huhns 1989]

[Gatrell 1984]

[Gause & Weinberg 1989]
[Gerson 1976]

[Gerson & Star 1986]

[Giddens 1976]

[Goffman 1971]

[Goffman 1974]

[Goffman 1979]

[Goffman 1981]

[Goguen 1990a]

[Goguen 1990b]

[Goguen 1993]

[Goguen & Linde 1993]
[Goldfarb 1981]

[Golipour-Koujali 1992]

[Goodwin & Heritage 1990]

[Gotel 1992]

[Gotel & Finkelstein 1993]

[Gotel & Finkelstein 1994]

[Granovetter 1974]

[Greenspan 1984]

[Greenspan et al. 1994]
[Grimes 1975]

[Grosz & Sidner 1986]

[Grudin 1994]

[Gumperz 1982]

[Hahn et al. 1991]

[Halliday & Hasan 1976]

[Hamblin 1971]

[Hamilton & Beeby 1991]

[Hammersley & Atkinson 1983]

[Harari 1979]

[Hares 1992]
References

[Harker et al. 1993]

[Harrington & Rondeau 1993]

[Heath 1982]

[Henderson-Sellers & Edwards 1990]

[Hewitt 1986]

[Hewitt 1991]

[Hirschheim & Klein 1989]

[Hirschheim et al. 1991]

[Hobbs 1979]

[Hoey 1983]
[Hoffman 1990]

[Hoffnagle & Beregi 1985]

[Horak 1985]

[Horwitz & Williamson 1986a]

[Horwitz & Williamson 1986b]

[Hughes 1984]

[Hughes et al. 1995]

[Huhns 1987]

[Humphrey 1988]

[Hymes 1972a]
[Hymes 1972b]

[IDE 1991]

[IEE 1991]

[IEE 1994]

[IEEE 1977]

[IEEE 1984]

[IEEE 1993]

[IEEE 1994]

[IEEE 1995]
[ISO 1986]  

[ISO 1989]  

[ISO 1991]  

[Jackson 1972]  

[Jackson 1991]  

[Jackson 1994]  

[Jackson 1995a]  

[Jackson 1995b]  

[James 1994]  
[Jarke & Pohl 1992]

[Jarke et al. 1993a]

[Jarke et al. 1993b]

[Jarke et al. 1994a]

[Jarke et al. 1994b]

[Jennings 1993]

[Jirootka 1991]
[Jirotka & Goguen 1994]

[Jirotka et al. 1992]

[Jirotka et al. 1995]

[Johnson & Johnson 1991]

[Johnson 1992]

[Johnson et al. 1984]

[Johnson et al. 1988]

[Johnson et al. 1991]

[Johnson et al. 1992]

[Jones 1994]
References

[Jones & Brooks 1994]

[Kahn & Kram 1994]

[Kaindl 1993]

[Kaiser et al. 1987]

[Kaplan 1990]

[Kavakli 1995]

[Keller et al. 1990]

[Kendon 1990]

[Keys 1991]
References

[Kling 1980]

[Knott & Dale 1993]

[Kunz & Rittel 1970]

[Kuwana & Herbsleb 1993]

[Laitinen 1992]

[Lange et al. 1994]

[Langford 1991]

[Latour 1987]

[Latour & Woolgar 1979]

[Laubengayer & Spearman 1994]
[Law 1991]

[Law 1994]

[Lee 1990]

[Lee & Lai 1991]

[Lefering 1993]

[Lefering 1994]

[Leite & Oliveira 1995]

[Levinson 1983]

[Levinson 1988]

[Lehman & Belady 1985]

[Lievegoed 1973]
[Linton 1936]

[Litterer 1963]

[Liu & Horowitz 1988]

[Liu & Horowitz 1989]

[Loka 1992].

[Loucopoulos & Karakostas 1995]

[Lowe 1985]

[Lubars et al. 1993]

[Luqi & Goguen 1993]
[Lutz 1993]

[Luff et al. 1993]

[Macfarlane & Reilly 1995]

[Mackenzie 1981]

[MacLean et al. 1991]

[Madhavji 1992]

[Mann 1984]

[Mann & Thompson 1983]
References

[Mann & Thompson 1988]

[March 1991]

[Martin & Powell 1992]

[Mathews & Ryan 1989]

[Matthews 1965]

[Mays et al. 1985]

[McDermid et al. 1994]

[McKeown 1985]

[Mi & Scacchi 1990]
[Mi & Scacchi 1992]  

[Microsoft 1993]  

[Moreno *et al.* 1994]  

[Morgan 1986]  

[Morgan 1990]  

[Morgan *et al.* 1983]  

[Morris & Coombes 1994]  

[Moser & Kalton 1971]  

[MST 1992]  

[MST 1993]  


[Palmer & Fields 1992]

[Parnas & Clements 1986]

[Parnas et al. 1985]

[Pickering 1992]

[Pohl 1994]

[Pohl & Jacobs 1994]

[Pohl et al. 1994a]

[Pohl et al. 1994b]

[Polack 1990]

[Potts et al. 1994]
References

[Pugh et al. 1971]

[Pyle et al. 1993]

[QED 1989]

[QSS 1994]

[Raffel 1979]

[Ramamoorthy et al. 1984]

[Ramamoorthy et al. 1986]

[Ramamoorthy et al. 1988]

[Ramesh & Dhar 1992]

[Ramesh & Edwards 1993]
[Ramesh et al. 1995]

[Reinhardt 1994]

[RESG 1995]

[Reubenstein & Waters 1991]

[Robinson 1991]

[Rolland 1994a]

[Rolland 1994b]

[Roman 1985]

[Rose 1993]

[Ross & Schoman 1977]
[Royce 1970]

[Ruddock 1969]

[Ryan 1995]

[Rzepka & Ohno 1985]

[Sacks 1995]

[SACO 1994]

[Said 1979]

[Salancik 1977]

[Saville-Troike 1982]

[Scott 1991]

[Seachrist 1995]
[Searle 1969]

[Shannon & Weaver 1949]

[Short 1988]

[Silverman 1970]

[SMI 1990]

[Smithers et al. 1991]

[Sodhi 1991]

[Sodhi 1992]

[Sol & Crosslin 1992]

[Sommerville 1989]
[Sommerville et al. 1993]

[Spurr et al. 1993]

[SSSL 1995a]

[SSSL 1995b]

[Star 1989]

[Stehle 1990]

[Strens & Dobson 1992]

[Stewart & Shamdasani 1990]

[Stillinger 1991]
References

[Stubbs 1983]

[Sumner 1992]

[Sushil & Raghunathan 1994]

[Swartout 1983]

[Sykes 1978]

[Takeda et al. 1993]

[Tannen 1989]

[Tannen 1993a]

[Tannen 1993b]

[TD 1995]
References

[Terry 1993]

[TI 1988]

[Thayer & Dorfman 1990]

[van Lamsweerde et al. 1990]

[van Lamsweerde et al. 1995]

[Waddington & Johnson 1989]

[Walton 1992]

[Ward 1992]

[Watkins & Neal 1994]

[Welsh & Han 1994]
[West 1991]

[Wheeler 1969]

[Wiener 1993]

[Wilmot 1992]

[Wing 1990]

[Winograd & Flores 1986]

[Winograd 1987/8]

[Wolfe 1994]

[Wood et al. 1994]
[Wright 1991]

[Wynekoop 1992]

[Yeh & Ng 1990]

[Yu 1993]

[Yu 1994]

[Yu & Mylopoulos 1994]

[Zave 1995]