Semantically Annotated Multi-Protocol Adapter Nodes: A New Approach to Implementing Network-Based Information Systems Using Ontologies

Nickolas John Gowland Falkner

The School of Computer Science
The University of Adelaide

October 30, 2007
Contents

1 Introduction 15
  1.1 Computing Resources ................................. 16
  1.2 Distributed Resource Usage ............................. 19
  1.3 Network-based Resources ............................... 19
    1.3.1 The Domain Name System ......................... 20
    1.3.2 The Network Time Protocol ....................... 20
  1.4 Large-scale network-based information services ............... 21
  1.5 Standards-based distributed systems ..................... 22
  1.6 User Requirements .................................. 23
  1.7 Global and Local Requirements ....................... 24
  1.8 The requirement for semantic alignment .................. 25

2 Literature Review 28
  2.1 Introduction ......................................... 28
    2.1.1 Data, Information and Knowledge .................. 29
    2.1.2 Chapter organisation .............................. 29
  2.2 Data ................................................ 30
    2.2.1 Operating Systems ................................ 31
    2.2.2 File systems and distributed file systems ........ 32
    2.2.3 Databases and Database Views ..................... 34
    2.2.4 Data storage and retrieval across networks ........ 35
      2.2.4.1 Network-based file systems .................... 35
      2.2.4.2 Potential issues with network-based file systems . 36
    2.2.5 Distributed Systems .............................. 38
      2.2.5.1 Definitions .................................. 38
      2.2.5.2 Models for distributed systems ................. 39
      2.2.5.3 Developing Distributed Systems with proprietary formats . 40
      2.2.5.4 Moving to open development frameworks .......... 41
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.2.5.5</td>
<td>Choosing the right design</td>
<td>42</td>
</tr>
<tr>
<td>2.3</td>
<td>Information</td>
<td>43</td>
</tr>
<tr>
<td>2.3.1</td>
<td>Network-Based Information Systems</td>
<td>43</td>
</tr>
<tr>
<td>2.3.1.1</td>
<td>Characterising a network-based information system</td>
<td>44</td>
</tr>
<tr>
<td>2.3.2</td>
<td>The Boundary of the Operating System</td>
<td>46</td>
</tr>
<tr>
<td>2.3.2.1</td>
<td>Specifying the boundary</td>
<td>47</td>
</tr>
<tr>
<td>2.3.2.2</td>
<td>The impact of policy-based control on system operation</td>
<td>48</td>
</tr>
<tr>
<td>2.3.3</td>
<td>The Syntax and Semantics of Programming Languages</td>
<td>49</td>
</tr>
<tr>
<td>2.3.4</td>
<td>Information Storage and the use of Associated Metadata</td>
<td>52</td>
</tr>
<tr>
<td>2.3.4.1</td>
<td>The Semantic File System</td>
<td>53</td>
</tr>
<tr>
<td>2.3.5</td>
<td>XML, RDF and Metadata</td>
<td>53</td>
</tr>
<tr>
<td>2.3.5.1</td>
<td>Metadata and Provenance</td>
<td>55</td>
</tr>
<tr>
<td>2.3.5.2</td>
<td>Existing work on provenance</td>
<td>56</td>
</tr>
<tr>
<td>2.3.6</td>
<td>Assembling Computational Resources</td>
<td>57</td>
</tr>
<tr>
<td>2.3.6.1</td>
<td>Workflow systems</td>
<td>58</td>
</tr>
<tr>
<td>2.3.6.2</td>
<td>The Grid</td>
<td>58</td>
</tr>
<tr>
<td>2.3.6.3</td>
<td>Organisational Structures</td>
<td>60</td>
</tr>
<tr>
<td>2.3.6.4</td>
<td>Virtual organisations</td>
<td>60</td>
</tr>
<tr>
<td>2.4</td>
<td>Knowledge</td>
<td>61</td>
</tr>
<tr>
<td>2.4.1</td>
<td>Impact of metadata on information storage</td>
<td>62</td>
</tr>
<tr>
<td>2.4.2</td>
<td>Required Logic</td>
<td>63</td>
</tr>
<tr>
<td>2.4.3</td>
<td>Description Logics</td>
<td>66</td>
</tr>
<tr>
<td>2.4.4</td>
<td>Ontologies</td>
<td>67</td>
</tr>
<tr>
<td>2.4.4.1</td>
<td>Providing context for data</td>
<td>70</td>
</tr>
<tr>
<td>2.4.5</td>
<td>Semantic web and web services</td>
<td>71</td>
</tr>
<tr>
<td>2.4.5.1</td>
<td>Semantic mark-up for web services</td>
<td>72</td>
</tr>
<tr>
<td>2.4.5.2</td>
<td>Semantic Annotation for WSDL</td>
<td>72</td>
</tr>
<tr>
<td>2.4.5.3</td>
<td>OWL-S</td>
<td>73</td>
</tr>
<tr>
<td>2.4.6</td>
<td>The Semantic Grid</td>
<td>74</td>
</tr>
<tr>
<td>2.4.7</td>
<td>A modern workflow-based system: myGrid</td>
<td>76</td>
</tr>
<tr>
<td>2.4.8</td>
<td>A legacy distributed system: The Domain Name System</td>
<td>77</td>
</tr>
<tr>
<td>2.4.8.1</td>
<td>A brief description of the DNS</td>
<td>77</td>
</tr>
<tr>
<td>2.4.8.2</td>
<td>Basic DNS elements</td>
<td>78</td>
</tr>
<tr>
<td>2.4.8.3</td>
<td>The operation of the DNS</td>
<td>78</td>
</tr>
<tr>
<td>2.4.9</td>
<td>Shortfalls</td>
<td>80</td>
</tr>
</tbody>
</table>
3 Challenges to implementing a network-based information system

3.1 Addressing the challenges to implementing a network-based information system

3.1.1 Heterogeneity

3.1.2 Public Network

3.1.2.1 The network as black box

3.1.3 Non-ownership

3.1.4 Non-local storage

3.1.5 Protocols and policies

3.2 Summary

4 Methodology

4.1 Developing a New Approach to Implementing NBIS

4.1.1 Knowledge Representation

4.1.1.1 Mechanisms for Representing Knowledge

4.1.1.2 Syntactic and Semantic Elements of Knowledge Representation Mechanisms

4.1.1.3 High-Availability and Remote Access Mechanisms

4.1.2 Context

4.1.3 Data, Information and Knowledge

4.1.4 A three-tiered software architecture

4.1.4.1 Ontological Specification Tier

4.1.4.2 Data Model Tier

4.1.4.3 Programming Language Semantics Tier

4.1.4.4 Encapsulation in the software architecture

4.2 The Nature of the Knowledge Contained in a System

4.2.1 Contextual Inference in Traditional Programming Systems

4.2.2 What constitutes the knowledge of a target system?

4.2.3 Modelling a system ontologically

4.2.4 An example of ontological capture in the software architecture

4.2.5 Choosing the correct level of metadata

5 Application of the Methodology

5.1 Target systems

5.1.1 Evolution

5.1.2 Data Exploitation
5.1.3 Local Specialisation ........................................... 119
5.1.4 Well-defined Purpose .......................................... 121
5.1.5 Metadata ......................................................... 122

5.2 Comparison with Existing Approaches .......................... 122
5.2.1 Workflow-based approach ...................................... 123
5.2.2 Orchestrative and Choreographic Approaches .................. 124
5.2.3 Service Oriented Architectures (SOA) .......................... 125
5.2.4 ACID-based transaction model ................................. 126

5.3 How the three-tier software architecture and KR-based approach meet the shortfalls ........................................... 129
5.3.1 Optimising data handling ........................................ 130
5.3.2 Simple versus Complex System Nodes .......................... 131

5.4 Modelling Program Operation and Knowledge Domains ....... 132
5.4.1 Aspects Common to All Distributed Systems ................... 133
  5.4.1.1 Process Management .................................. 134
  5.4.1.2 Input/Output ........................................... 134
  5.4.1.3 Concealment of implementation ........................ 135
  5.4.1.4 The use of knowledge in the system ...................... 135
5.4.2 Distinct Aspects of Different Distributed Systems .......... 137
5.4.3 Modelling operating systems and distributed information systems .... 138
5.4.4 Fundamental Functions ......................................... 141
5.4.5 Remotely Installable Native Functions ........................ 144
5.4.6 Fitting into the model .......................................... 146

6 Conceptual Model ................................................. 147
6.1 The Problem Domain .............................................. 147
6.2 High-level Conceptual Model ...................................... 149
  6.2.1 Query formats and the store ................................ 153
    6.2.1.1 Terminal and non-terminal graph nodes ............... 154
    6.2.1.2 Summary of query operations .......................... 155
  6.2.2 Operational Semantics ..................................... 155
  6.2.3 Object Conceptualisation ................................... 157
6.3 Formal and Theoretical Basis: RDF and OWL .................... 157
  6.3.1 RDF ......................................................... 159
    6.3.1.1 Blank Nodes and Interpretations ..................... 161
    6.3.1.2 Entailment ........................................... 161
6.3.2 OWL ............................................................... 162
   6.3.2.1 Entailment in OWL and Decidability ................. 162
   6.3.2.2 Significance ........................................... 163
6.3.3 Ontological grounding in RDF and OWL semantics .......... 164
   6.3.3.1 Entailment of operational semantics ............... 164
   6.3.3.2 Is entailment required? .............................. 166
6.4 Summary of Methodology ........................................ 168

7 Implementation .................................................. 170
7.1 Pragmatic Requirements of the Model .......................... 171
   7.1.1 Top level structure of the ontology ...................... 172
   7.1.2 Lower level structure of the Functional Branch .......... 174
   7.1.3 Data Model Issues ....................................... 175
      7.1.3.1 Type Information ..................................... 175
      7.1.3.2 Derived Datatypes ................................. 176
      7.1.3.3 Introducing new types ............................... 177
7.2 Choosing a target system ...................................... 178
7.3 The Domain Name System (DNS) ................................ 179
   7.3.1 Operational Issues in the DNS .......................... 180
   7.3.2 The effect of strict control on system modification .... 181
   7.3.3 Enforced configuration versus evolutionary configuration 181
   7.3.4 The DNS and semantic enhancement technologies ........ 183
7.4 The two implementations of the DNS in a partial and more developed KR framework .......................................... 184
7.5 Problems in self-definition ................................... 184
7.6 Developing a semantic model of the DNS ....................... 185
7.7 Mapping the DNS into the model ................................ 186
   7.7.1 Mapping into the Ontology and Data Model ............... 187
7.8 Describing the operational semantics .......................... 188
   7.8.1 A language for capturing operational semantics .......... 191
   7.8.2 Language Choice ......................................... 192
7.9 Ontological development of the Operational Semantics sub-branch ......................................................... 192
7.10 Native Functions ................................................ 195
7.11 Application: Extending the functionality of Internet services ............................................................. 196
   7.11.1 Extending an existing service ............................ 196
7.12 Implementation of an ontology based DNS server ............. 199
7.13 Ontological Reasoning and Entailment ................................................. 203
7.14 Policy specifications ................................................................. 206
7.15 Aggregating metadata and data .................................................. 207
7.16 Ontology support in the client .................................................... 209
7.17 Summary ..................................................................................... 210

8 Results ............................................................................................. 211
8.1 Experimental Outline ...................................................................... 211
  8.1.1 DNS structure ........................................................................... 212
8.2 Testbed Environment ........................................................................ 213
8.3 $\alpha$-system experiments ................................................................. 214
  8.3.1 Time Complexity of Queries ..................................................... 217
8.4 $\beta$-system experiments ................................................................. 218
  8.4.1 Multi-protocol server ................................................................. 218
8.5 Breakdown of performance components in the $\beta$-implementation ....... 221
  8.5.1 Experiment 2-1 ......................................................................... 223
    8.5.1.1 Analysis of Experiment 2-1 results ...................................... 224
  8.5.2 Experiment 2-2 ......................................................................... 226
    8.5.2.1 Complexity rating for ontologies ...................................... 226
    8.5.2.2 Test Ontologies for Experiment 2-2 ................................. 227
    8.5.2.3 Analysis of Experiment 2-2 results ...................................... 227
  8.5.3 Subsequent Experiments with Faster Equipment ......................... 230
    8.5.3.1 Experiment 2-1-N results .................................................. 231
    8.5.3.2 Experiment 2-2-N results .................................................. 231
    8.5.3.3 Analysis of new performance results ................................... 233
  8.5.4 Further optimisation .................................................................... 233
    8.5.4.1 Experiment 2-1-N-C results .............................................. 234
    8.5.4.2 Experiment 2-2-N-C results .............................................. 234
    8.5.4.3 Analysis of results of the optimised code experiments ........ 234
8.5.5 Complexity of queries ................................................................. 235
8.5.6 Overhead due to multi-protocol issues ......................................... 236
8.5.7 Performance bottlenecks ............................................................. 238
8.5.8 Optimisation points ..................................................................... 238
8.6 Summary ......................................................................................... 238
9 Future Work and Applications 240

9.1 Extensions .................................................. 240
9.1.1 Operational Semantics .................................. 240
9.1.2 Representation of native functions .................. 241
9.1.3 Self-healing mechanisms ............................... 242
9.1.4 Confidence-based protocol determination .......... 243
9.1.5 Ontology repair for DL features ..................... 244
9.1.6 Name abstraction ....................................... 245
9.1.7 Denotational Semantics ............................... 245

9.2 Applications .................................................. 246
9.2.1 Interoperation with WSDL ............................ 246
9.2.2 Interoperation with SAWSDL ......................... 247
9.2.3 Interoperation with OWL-S ........................... 247
9.2.4 Semantic Web .......................................... 248
9.2.5 General distributed systems ......................... 249
9.2.6 Grid computing ......................................... 251

9.3 Summary ..................................................... 252

10 Conclusions 253

10.1 Specifying Semantics ................................. 254
10.2 Contribution .............................................. 255
10.3 Meeting User Expectations of a Distributed Information System .......... 256
10.3.1 Meeting Global and Local Requirements .......... 257
10.3.2 Performance Implications .......................... 258
10.4 The Advantages of a Knowledge-Based Approach .......... 259
10.4.1 Explicit Ontological Representation Minimises Confusion .......... 259
10.4.1.1 Shared Concepts Allows Much Wider Reuse of Data ........ 260
10.4.2 Reduced Double-Handling of Data ................. 260
10.4.3 Increased Opportunities for Collaboration ............ 261
10.4.4 Decoupling of Stored Data from External Representations .......... 262

A The core ontology for DNS operations 264


List of Figures

1.1 A simple Java program .............................................. 17
2.1 Dublin Core Metadata ............................................. 57
2.2 Dublin Core metadata in HTML ................................. 57
2.3 Dublin Core metadata in RDF [61] ............................ 58
2.4 An example of OWL DL .......................................... 70
2.5 An example of KIF preconditions [98] ......................... 74
2.6 myGrid services and middleware stack from [64] .......... 76
4.1 Block structure of an ontologically-defined NBIS node .... 104
4.2 Encapsulation in the three-tier software architecture ...... 108
4.3 The three branch ontology ...................................... 112
5.1 The three-tier software architecture, with an operating system mapping .... 136
5.2 Fundamental function sets based on system requirements .... 142
6.1 The problem domain for widely-distributed network-based information systems .............................. 148
6.2 A network-based information system .......................... 148
6.3 The high-level functional conceptual representation of a network-based information system server .................. 150
6.4 The high-level functional conceptual representation of a network-based information system client .................. 150
6.5 The high-level functional conceptual representation with context establishment .................................... 152
6.6 The high-level functional conceptual representation showing detail of the ‘Act’ processes ................................. 153
6.7 A UML Sequence Diagram for the Object Conceptualisation .... 158
6.8 An RDF Graph showing urirefs and literals, connected by arcs. ......................... 160
6.9 The RDF Graph from Figure 6.8 shown in RDF/XML .......... 160
6.10 Example graph form of a Lambda expression ........................................ 165
7.1 The minimal base ontology that maintains strong context ......................... 173
7.2 The location of traditional DNS semantics in the UML sequence model .... 188
7.3 The pre-implementation model of the DNS with operational semantics .... 189
7.4 The minimal base ontology with DNS behaviour overlaid ....................... 190
7.5 The location of the fixed point. ............................................................ 193
7.6 Using specifications to encapsulate data for transport to the client .......... 194
7.7 An example of the applied $\lambda$-calculus. ........................................... 195
7.8 The post-implementation model of the DNS with operational semantics ... 201
7.9 The operational semantics component, showing cryptographic functions ... 202
7.10 A simple DNS query illustrated. .......................................................... 207
7.11 A multi-protocol response to a simple DNS query .............................. 208
7.12 A metadata-rich response to a DNS query from an enhanced system ...... 208
8.1 An example DNS structure showing the location of ontologically enhanced
    nodes .................................................................................................. 213
8.2 The delegation of name domains to servers in the DNS ......................... 214
8.3 Multi-format service on separate ports. ............................................... 220
8.4 Multi-format service on a shared port. ............................................... 220
8.5 The graph of complexity metric versus relative store size .................... 229
8.6 The graph of relative store size versus TREE and LOAD times ............ 230
8.7 The graph of complexity metric versus TREE and LOAD times ............ 231
9.1 Adapter nodes on the boundary displayed against adapter nodes without
    occlusion ............................................................................................ 250
9.2 Adapter nodes on the boundary displayed against adapter nodes as occluders 250
10.1 The expected deployment of adapter nodes versus standard nodes ......... 259
List of Tables

2.1 A Classification of Distributed Systems ........................................... 44
8.1 Performance measurements ......................................................... 215
8.2 Client-side Performance measurements with ontology load .................. 219
8.3 Server-side Performance measurements without ontology load ............... 219
8.4 Client and server operations ....................................................... 221
8.5 Results of experiment 2-1, showing time spent in each activity. .............. 224
8.6 Test ontologies for experiment 2-2 showing complexity metric ............... 228
8.7 LOAD and TREE times for the ontologies in experiment 2-2. ............... 228
8.8 Results of experiment 2-1-N, showing time spent in each activity. .......... 232
8.9 LOAD and TREE times for the ontologies in experiment 2-2-N. ............. 232
8.10 Results of experiment 2-1-N-C, showing time spent in each activity. ....... 234
8.11 LOAD and TREE times for the ontologies in experiment 2-2-N-C. .......... 234
Abstract

Network-based information systems are an important class of distributed systems that serve large and diverse user communities with information and essential network services. Centrally defined standards for interoperation and information exchange ensure that any required functionality is provided but do so at the expense of flexibility and ease of system evolution. This thesis presents a novel approach to implementing network-based information systems in a knowledge-representation-based format using an ontological description of the service. Our approach allows us to provide flexible distributed systems that can conform to global standards while still allowing local developments and protocol extensions.

We can share data between systems if we provide an explicit specification of the relationship between the knowledge in the system and the structure and nature of the values shared between systems. Existing distributed systems may share data based on the values and structures of that data but we go beyond syntax-based value exchange to introduce a semantically-based exchange of knowledge. The explicit statement of the semantics and syntax of the system in a machine-interpretable form provides the automated integration of different systems through the use of adapter nodes. Adapter nodes are members of more than one system and seamlessly transport data between the systems.

We develop a multi-tier software architecture that characterises the values held inside the system depending on an ontological classification of their structure and context to allow the definition of values in terms of the knowledge that they represent. Initially, received values are viewed as data, with no structural information. Structural and type information, and the context of the value can now be associated with it through the use of ontologies, leading to a value-form referred to as knowledge: a value that is structurally and contextually rich. This is demonstrated through an implementation process employing RDF, OWL and SPARQL to develop an ontological description of a network-based information system. The implementation provides evidence for the benefits and costs of representing a system in such a manner, including a complexity-based analysis of system performance.

The implementation demonstrates the ability of such a representation to separate global standards-based requirements from local user requirements. This allows the addition of
behaviour, specific to local needs, to otherwise global systems in a way that does not compromise the global standards.

Our contribution is in providing a means for network-based information systems to retain the benefits of their global interaction while still allowing local customisation to meet the user expectations. This thesis presents a novel use of ontologically-based representation and tools to demonstrate the benefits of the multi-tier software architecture with a separation of the contents of the system into data, information and knowledge. Our approach increases the ease of interoperation for large-scale distributed systems and facilitates the development of systems that can adapt to local requirements while retaining their wider interoperability. Further, our approach provides a strong contextual framework to ground concepts in the system and also supports the amalgamation of data from many sources to provide rich and extensible network-based information system.
Declaration

This work contains no material that has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text.

I give consent to this copy of my thesis, when deposited in the University Library, being available for loan and photocopying.

Signed:

Nickolas John Gowland Falkner
Acknowledgements

My thanks go to my supervisors, Dr Paul Coddington and Dr Andrew Wendelborn, as they have both helped me greatly with their advice, effort and ongoing support. Their insight and suggestions have been invaluable and I am a better researcher for it.

My peers, among the postgraduate students, and, later, also among the staff, academic and professional, have been of great help with their ongoing comments, assistance and willingness to enter into discussions. Associate Professor David Munro shared a number of his experiences from his journey into Academia and provided me with some much-needed confidence-boosting at points where I was doubting myself and the journey as a whole.

I would also like to thank my mother for the love, support and educational background with which she provided me.

My thanks also to my friends, who have been very tolerant while I tried hard not to dominate simple ‘how are you’ queries with a detailed discussion of the problem of loose contextual frameworks. I didn’t always succeed but they keep inviting me to dinner.

Finally, I would like to thank my wife who has supported me with a great deal of love and good humour while I undertook this project. She has also helped with a great deal of good advice based on her own experience as a PhD student and as a hard-working academic. She has also shown quite amazing discretion and judgement in the way in which she shared her experience with me. This work would not have been completed without her.

Thank you all, very much.