



University
of Southampton

Improving the practice of Operations Support Systems in the Telecommunications Industry using Open Source

by

Craig Gallen

A Thesis submitted for the degree of
Doctor of Engineering

School of Electronics and Computer Science,
University of Southampton,
United Kingdom.

Date: November 2007

00396848

Table of Contents

Table of Contents 2
Document Notes 7
0.1 Course Details 7
0.2 Change Control 7
0.3 Abstract 7
0.4 Acknowledgments 7
0.5 List of Figures 8
0.6 List of Tables 9
Executive Summary 12
Chapter 1 Introduction to the Area of Research Interest 15
1.1 Background 15
1.2 Standards and Open Source 16
1.3 Brief history of Network Management leading up to NGOSS 18
1.3.1 Early work with TMN 18
1.3.2 CORBA and TINA-C Based Management 19
1.3.3 Technology Neutral OSS Component Architectures 20
1.3.4 TMForum NGOSS 21
1.3.5 TMForum NGOSS and DMTF WBEM 23
1.4 A Critique of OSS History 25
1.5 Perceived Problems with the Standards Development Process 26
1.6 Perceived Gaps in Network Management Research 28
1.7 The TMForum's contribution to industry knowledge 29
1.8 Summarising the Research Theme 30
1.9 Relevance of results 31
1.10 Organisation of the Thesis 31
Chapter 2 Research Methodology: Action Research 34
2.1 Defining Information Systems Action Research 34
2.1.1 Action Research Epistemology 35
2.1.2 Action Research Process 36
2.1.3 Research Themes 37
2.1.4 Entering the Problem Situation 37
2.1.5 Establishing Roles 38
2.1.6 Reflective Learning 38
2.1.7 Overall Learning Cycle 39
2.1.8 Quality in Action Research 39
2.2 Research Method used in this program 40
2.2.1 Participant Action Research 40
2.2.2 Learning Cycle 42
2.2.3 Entering the Problem Situation and Establishing Roles 44
2.2.4 Participative Action Research within the TMForum 46
2.2.5 Data 47
2.2.6 Ethics 47
2.3 Summary 49
Chapter 3 The OSS Industry and Open Source Problem Space 51
3.1 Chapter Context 51
3.2 Introduction 51
3.3 Risk Efficiency 51
3.3.1 Uncertainty Management 52
3.3.2 Better Investment Decisions 55
3.3.3 Summarising Open Source Uncertainty Management 59
3.4 OSS Component Market Place 59

3.4.1 Network Effects in the Telecoms Industry 59
3.4.2 Customers - Telecoms Service Providers 62
3.4.2.1 Outsourcing of Functions 62
3.4.2.2 Location of R&D Resources 63
3.4.2.3 Telecommunications Industry Layer Model 64
3.4.3 Component Developers 67
3.4.3.1 Telecoms Equipment Vendors 68
3.4.3.2 OSS Independent Software Vendors (ISV's) 70
3.4.3.3 Middleware Component Vendors 71
3.4.3.4 Specialist OSS Middleware Component Vendors 72
3.4.4 Component Assemblers 72
3.4.5 Summarising the OSS Component Market Place 74
3.5 Development Approach 74
3.5.1 Industry Strategy 74
3.5.1.1 Strategic Questions Concerning OSS Open Source 74
3.5.1.2 Sources of Strategic Uncertainty in the OSS Component Marketplace 75
3.5.1.3 NGOSS as a means for Strategic GCBA Risk Reduction 76
3.5.2 Tactics for a successful OSS project 78
3.5.2.1 Tactical Questions concerning OSS Open Source 78
3.5.2.2 Sources of Tactical Uncertainty in Open Source or COTS Based Development 79
3.6 Collaborative Innovation within the TMForum 80
3.6.1 The need to integrate R&D and Marketing in New Product Introduction 81
3.6.2 Role of the TMForum, Universities and other Non-Market Institutions 84
3.6.3 The Potential Role of Open Source in Innovation 86
3.7 Research Program Design 90
Chapter 4 Intervention: OpenOSS in the TMForum 94
4.1 Introduction 94
4.2 The OpenOSS Catalyst Project 94
4.2.1 Diagnosis 94
4.2.2 Action Planning 98
4.2.2.1 Expected Risk Profile of the OpenOSS Catalyst 98
4.2.2.2 Open Source Service Network 100
4.2.2.3 Open Source OSS Stack 101
4.2.3 Action Taking 102
4.2.4 Evaluating OpenOSS 104
4.2.4.1 Immediate Evaluation 104
4.2.4.2 Later Developments 106
4.3 Intervention: NGOSSoil Case Study 107
4.3.1 Diagnosis 107
4.3.2 Action Planning 107
4.3.3 Action Taking 109
4.3.4 Evaluating NGOSSoil 112
4.3.4.1 The Potential Significance of NGOSSoil to the wider OSS community 112
4.3.4.2 The Potential Significance of NGOSSoil to Nortel 114
4.3.4.3 Lessons Learnt from NGOSSoil 116
4.4 Related Action Research 118
4.4.1.1 R&D Net 118
4.4.1.2 University of Delft UMTS Test Bed 119
4.4.1.3 Process Portals 119
4.4.1.4 Action Research as Entrepreneurship 120
Chapter 5 Intervention: Digital TV Service Management 123
5.1 Introduction 123
5.2 Action Research and the NGOSS Lifecycle 123
5.3 Diagnosis 124
5.3.1 Industry Background 124

5.3.2 Managing SLA's in Telecommunications and Broadcasting 126
5.3.3 Arqiva's Requirements 127
5.4 Action Planning 128
5.4.1 Applying Telecommunications SLA Management Standards 129
5.4.1.1 SLA Definition 129
5.4.1.2 Identifying DVB-T service scenarios 129
5.4.1.3 Identifying DVB-T service topology 130
5.4.1.4 Developing an NGOSS System View of DVB-T Service Management 131
5.4.1.5 Resource Model of a DVB-T Network Topology 134
5.4.1.6 Modelling a Transport Stream 134
5.4.1.7 Modelling a DVB-T Network Topology 136
5.4.1.8 NGOSS System View of DVB-T Service Management 140
5.5 Action Taking 143
5.5.1 Trial NGOSS Implementation View 143
5.5.1.1 Trial Scope 143
5.6 OSS/J Messaging Protocol 146
5.6.1.1 Site Manager 146
5.6.1.2 Central Manager interface to StateWise 147
5.7 Evaluating the Digital TV Proof of Concept 147
5.7.1 Use of Standards 148
5.7.2 Cost Containment 148
5.7.3 Dissemination of Information and Innovation 149
5.8 Related Work 150
5.8.1 NGOSS DVB-S Digital Movie Distribution 150
5.8.2 Action research into information systems modelling 151
Chapter 6 Intervention: Contributing an OSS/J Interface to OpenNMS 154
6.1 Introduction 154
6.2 Diagnosis 154
6.2.1 Choosing To Engage With An Open Source Project 154
6.3 Action Planning 156
6.4 Action Taking 158
6.5 Evaluation 158
6.5.1 OpenNMS, Standards and Innovation 158
6.5.2 Reflective learning from adding OSS/J to OpenNMS 160
6.5.2.1 Initial Open Source Project Selection Process 160
6.5.2.2 The Technical Evaluation Process 161
6.5.2.3 The Value of Introducing Component Frameworks 162
6.5.3 Open Source as an aid to OSS Standards Implementation 165
6.5.4 An antidote to Fear Uncertainty and Doubt 166
6.5.4.1 Courage 166
6.5.4.2 Conviction 168
6.5.4.3 Faith 169
Chapter 7 Discussion and Conclusion 173
7.1 Summarising the Thesis 173
7.2 Reflections on the impact of open source on telecoms OSS (Mps) 175
7.2.1 NGOSS and Open Source 175
7.2.2 Impact on practice 179
7.2.2.1 Service Providers 179
7.2.2.2 ISV's 181
7.2.2.3 Integrators 183
7.2.2.4 Equipment Vendors 184
7.2.2.5 Standards Bodies 184
7.2.3 Risk Efficiency and Open Source Introduction 185
7.3 Reflections on the Action Research Process (Mr) 187
7.4 Future Work 190

7.5 Conclusion 190
References 193
Glossary 220
Appendix A Chronology of the Action Research Program 226
A.1 A Concise Project History 226
A.2 Detailed project chronology 228
Appendix B Meeting Invocom's Research Objectives 238
B.1 Eng D Research 238
B.2 Meeting Sponsor Stakeholder Expectations 238
B.3 Choosing an applicable Research Methodology 241
Appendix C Impact of Open Source on eTOM Processes 246
Appendix D Open OSS Stakeholder Expectations 252
D.1 ISV Stakeholder Expectations 252
D.2 Integrator Stakeholder Expectations 253
D.3 Non Market Institutions (Universities and TMForum) Stakeholder Expectations 254
D.4 Service Provider Stakeholder Expectations 255
D.4.1 BT's Network and Service Strategy 255
D.4.2 BT's Standards Strategy 259
D.4.3 BT's OSS Rationalisation Strategy 259
D.4.4 BT's NGOSS Modelling Strategy 261
D.4.5 BT's Open Source Research objectives 263
D.4.6 Other Service Provider Stakeholder Expectations 264
Appendix E OpenNMS Case Study 266
E.1 OpenNMS History 266
7.6 OpenNMS as a company within an Open Source Cluster 268
E.2 OpenNMS Users 269
E.3 OpenNMS Organisation and Developers 271
E.4 OpenNMS Design 272
E.4.1 OpenNMS Server 272
E.4.1.1 Network Management Interface Daemons 274
E.4.1.2 Correlation and Work Flow Daemons 275
E.4.1.3 External Interfaces 276
E.4.1.4 Persistent Storage 276
E.4.1.5 Spring Framework 277
E.4.2 OpenNMS Web Application 277
E.4.3 OpenNMS Development Environment and Process 278
E.4.3.1 Test Driven Development 278
E.4.3.2 Maven 2 project build system 278
E.4.3.3 Code Management and Developer Communications 281
E.5 Innovation in the OpenNMS Project 281
E.5.1 Innovation in the OpenNMS Business Model 281
E.5.2 Innovation in core technologies 283
E.5.3 OpenNMS Maps Innovation 283
E.5.4 OpenNMS Correlation Innovation 285
E.5.5 User Innovation in Network Management around OpenNMS 286
Appendix F OpenNMS OSS/J interface Design 289
F.1 Project Chronology 289
F.2 Project Overview 292
F.2.1 Introduction to OSS/J 292
F.2.2 Design Philosophy 293
F.2.3 OSS/J Conformance 294
F.2.4 Production Use 295
F.2.5 Licence 295
F.3 OSSBeans Project 295

F.4 OpenNMS qosdaemon 299
F.4.1 Functionality Provided 299
F.4.2 QoSD daemon 299
F.4.2.1 Native OpenNMS provided interface. 299
F.4.2.2 Separate J2EE server provided interface. 300
F.4.3 QoSDrx 301
F.5 OpenNMS qosdaemon Package Description 302
F.5.1 Functionality Overview 302
F.5.2 Package Overview 302
F.5.2.1 OssDao 302
F.5.2.2 QoSD Daemon 303
F.5.2.3 QoSDrx Deamon 306
Appendix G OpenOSS Catalyst Final Technical Report 310
G.1 Executive Overview 311
G.2 Introduction 312
G.3 Project Objectives 312
G.4 Collaborative Environment 313
G.5 Test Network 314
G.5.1 Network Scenario 314
G.5.2 Shared remote access lab 315
G.5.3 The test network 315
G.5.4 VoIP 317
G.6 Management Solution 319
G.6.1 Solution overview 319
G.6.2 OpenOSS Catalyst realisation of the OSS stack 321
G.6.3 VoIP Management (Agilent) 322
G.6.4 Performance Management (BUTE) 324
G.6.5 The Web Interface 326
G.6.6 Fault Management & Correlation (Southampton) 327
G.7 Experiences 333
G.8 Next Steps 334
G.9 Conclusions 335
G.10 References 335
G.11 Document History 335

Document Notes

0.1 Course Details

Course	Engineering Doctorate (Eng D)
University Supervisor	Dr. Jeff Reeve
Industrial Supervisor	Mr Simon Pearson
Research Engineer	Craig Gallen CEng MIEE MBA MSc. Bsc Student ID - 02233568

0.2 Change Control

Version	Filename	Status	Author	Date
1.0	C Gallen Eng D Thesis 1.0 11-12-07.doc	Submitted	Craig Gallen	11 Dec. 07
1.1	C Gallen Eng D Thesis 1.1 16-03-08.doc	Final	Craig Gallen	16 Mar. 08

0.3 Abstract

This document fulfils the requirements for the final Thesis in completion of an Engineering Doctorate program at the University of Southampton.

This Thesis will present the findings of a three and a half year Engineering Doctorate study (2003-2007) which investigated the options for open source in delivering Operational Support Systems (OSS) to the global telecommunications industry. The principal mode of investigation has been Action Research in collaboration with a number of industry players and under the auspices of the Telemanagement Forum. The principal contribution has been to develop within the industry a greater understanding and acceptance of the role which open source can have in the delivery of OSS solutions.

Keywords

Open Source, NGOSS, OSS/J, Action Research, OSS, Operations Support Systems, Systems Integration, Requirements Management

0.4 Acknowledgments

I would like to thank Invocom for their direct sponsorship of this Eng D Research. In particular, I would like to thank my supervisors Dr Jeff Reeve and Mr Simon Pearson and Mr Eric Brown for their support and encouragement throughout.

Action Research is a collaborative process and none of this work could have been achieved without the help of an army of people. I would especially like to thank;

- Simon Pearson, Eric Brown, Paolo Imbalzano and the other Invocom employees who offered help and advice.
- Gary Bruce, John Wittgreffe, Paul Robson Nektarios Georgalas and Manooch Azmoodeh from BT
- Andrew Chalmers, Martin Creaner, Debbie Burkett and the staff of the TMForum
- Graham Calvert, Paul Sherratt, Colin Ashford, Chris Hobbs and all the other NGOSSoil contributors from Nortel.
- Tarus Balog, Dave Hustace, Matt Brozowski, Jonathan Sartin and all the other OpenNMS developers
- Peter Mottishaw and Doug Carson from Agilent
- Bob Potter, Martin Johnston, Kathryn Ellis and Chris Ash from Arqiva Communications
- Martin Hobbs and Don Keir of Sidonis
- Cliff Faurer from Automagic KB
- Christophe Ebro and Philippe Lalande from Sun Microsystems & OSS/J
- Sravan Desikan from Cognizant

And last, but not least

- Ricky Spaven, Tim Griffith, Gavin Willingham, Nick Dance, Paul Hickman and Simon Utting the long suffering Southampton University Students (Where are you now?)

As Always, the opinions, conclusions and any errors of judgement or omissions in this Thesis remain entirely the author's responsibility.

0.5 List of Figures

Figure 1 Action Research cycle 36
Figure 2 Revised Action Research Framework 36
Figure 3 Action Research Process 37
Figure 4 Summary of chronology of projects during the Action Research program 47
Figure 5 Risk Efficient options 53
Figure 6 Technology Adoption Curve and Moore's chasm 54
Figure 7 Simplified view of COTS based development 73
Figure 8 Communications gaps between functions in Vendors and Customers 81
Figure 9 The Innovation Community 85
Figure 10 Risk Profile of OpenOSS 99
Figure 11 VoIP Network Scenario 101
Figure 12 Components in OpenOSS catalyst 101
Figure 13 Nortel DML Functional Diagram 108
Figure 14 Simplified DVB-T network 131
Figure 15 SID Product Service Resource relationships 132
Figure 16 UML model of relationships between components of a DVB-T service 135
Figure 17 Modelling a DVB-T broadcast network in the SID 139
Figure 18 Illustrating how DVT-T MUX's and Transport Streams are modelled 140
Figure 19 Relationship between TAM functional blocks in DVB-T SLA management 141
Figure 20 DVB-T Management Trial Implementation 144
Figure 21 Data Model used in Proof of Concept 145
Figure 22 OpenNMS Stand-Alone Features vs. OSS Requirements at Q4 2005 157
Figure 23 Proposed OpenNMS OSS Evolution 157
Figure 24 Illustrating Dependencies between Applications, Frameworks and Component Projects 164
Figure 25 Conceptual Model for Enterprise Adoption of Open Source SW 187
Figure 26 eTOM Processes directly influencing COTS or open source procurement 246
Figure 27 eTOM Processes relationships directly influencing COTS or open source procurement 247
Figure 28 BT's Legacy Network layering 256
Figure 29 BT's proposed 21CN network 256
Figure 30 BT Physical NW Infrastructure 256
Figure 31 BT Example of SIP VoIP 258
Figure 32 BT Service Execution Framework 258
Figure 33 OpenNMS Maturity Curve 268
Figure 34 OpenNMS Architecture 273
Figure 35 Simplified OpenNMS Object Model 273
Figure 36 Original OpenNMS 1.0 Swing GUI 284
Figure 37 OpenNMS Italian Adventures SVG GUI 284
Figure 38 OSSBeans - compiling a marshalling framework from OSS/J XSD's 297
Figure 39 OSSBeans class diagram 298
Figure 40 Native OpenNMS provided interface 300
Figure 41 Separate J2EE server provided interface 301
Figure 42 Illustrating remote slave OpenNMS systems talking to a central master system 301
Figure 43 VoIP Network Scenario 314
Figure 44 Simulation Network 315
Figure 45 Nistnet 316
Figure 46 Distributed Internet Traffic Generator D-ITG 317
Figure 47 Routing of 'excess traffic' 317
Figure 48 VoIP Soft Phone 318
Figure 49 Routing of voice calls 318
Figure 50 Open OSS Catalyst mapping to eTOM 320
Figure 51 Conceptual OSS stack 321
Figure 52 Components in OpenOSS catalyst 322
Figure 53 Agilent architecture for OpenOSS 322

Figure 54 Agilent GUI for VoIP data 324
Figure 55 QoS portlet interaction diagram. 325
Figure 56 QoS PM Solution Design. 326
Figure 57 OpenNMS alarms prior to correlation 328
Figure 58 Correlated trouble ticket 328
Figure 59 Class diagram of Correlation solution 331
Figure 60 OTRS Architecture 331
Figure 61 OTRS integration diagram 332

0.6 List of Tables

Table 1 Results from this Action Research Program 13
Table 2 Papers and presentations 13
Table 3 Henning's reasons for the failure of market adoption of CORBA 25
Table 4 Perceived Benefits of TMForum Catalyst Program 30
Table 5 Organisation of this Thesis 32
Table 6 Action Research Activities 35
Table 7 Categories of Action Research 35
Table 8 Project Management Maturity Model For an IS action-research project 39
Table 9 A framework for rigour in Qualitative Action Research 40
Table 10 Analysis Framework 55
Table 11 Summarising Risk Efficiency Concepts 59
Table 12 OSS purchasing patterns of Telecoms operators 64
Table 13 Telecoms industry layer model and open source penetration 65
Table 14 Prominent telecoms specific open source projects 67
Table 15 Architectural level of COTS 68
Table 16 Summarising the OSS Component Market Place 74
Table 17 Categorisation of COTS based projects 76
Table 18 Managing frequent, continuous change in OSS COTS marketplace 78
Table 19 Differences between COTS and open source components risk profiles 80
Table 20 Project Risks Associated with Using Off The Shelf SW Components 80
Table 21 Summary of marketing and R&D roles 82
Table 22 Summary of Marketing and R&D Stereotypes 83
Table 23 Integrating Mechanisms to improve R&D/Marketing cooperation 83
Table 24 Potential effect of Open Source on NPI process 84
Table 25 OpenOSS Requirements Derived From Innovation Toolkit Requirements 90
Table 26 Key characteristics of each project intervention 91
Table 27 Open OSS Project Analysis Using 6 W's Analysis framework 96
Table 28 Risk characterisation of typical OSS projects 99
Table 29 Open source components used by Truphone 106
Table 30 Analysis framework NGOSSoil Project Analysis Using 6 W's 111
Table 31 Common Information Model (CIM) Repository Implementations 113
Table 32 NGOSS Lifecycle SANRR Methodology 124
Table 33 TMForum steps for developing Service level Agreements 129
Table 34 Simplified DVB-T Service Scenarios 130
Table 35 DVB-T SLA management package descriptions 142
Table 36 Requirements for Arqiva Proof of concept NGOSS Implementation View 146
Table 37 Open source components used by Cast4all 150
Table 38 Criteria For Choosing an Open Source OSS Project to Research 156
Table 39 Categorisation of Enterprise systems 187
Table 40 Strategic Recommendations for Invocom 241
Table 41 computing research methods 241
Table 42 Definitions of research methodologies 242
Table 43 eTOM process areas potentially affected by the introduction of Open Source 248
Table 44 OSS capabilities required for next generation services 259

Table 45 BT's Reusable Engineering Capabilities 261
Table 46 Major COTS Vendors in the BT's OSS Implementation 261
Table 47 Key OpenNMS Milestones 266
Table 48 Example OpenNMS Users Within The Telecoms Industry Layer Model 270
Table 49 Roles of OpenNMS Group employees 272
Table 50 Open source components used by OpenNMS 279
Table 51 Other Open Source Projects interacting with OpenNMS 286
Table 52 OpenOSS development milestones and overlapping OpenNMS milestones 291
Table 53 Components used to create test network 318
Table 54 COTS components used in the proof of concept 324

Executive Summary

Executive Summary

Industry analysts Gartner predict that by 2011 at least 80% of all commercial software solutions will include substantive open source (*Driver, M. 2007*). How to respond to the open source challenge is becoming a major consideration for all managers involved with IT. This Thesis will present the findings of a three and a half year Engineering Doctorate study (2003-2007) which investigated the options for open source in delivering Operational Support Systems (OSS) to the global telecommunications industry.

The EPSRC Engineering Doctorate best practice guidelines (*EPSRC 2006*) note that EngD programs are intended to provide a mechanism and framework for high quality collaboration between academic groups and a range of companies in order to contribute to the body of knowledge on a particular technical discipline, industrial sector or multidisciplinary theme. The foremost body representing the interests of the OSS industry is the Telemanagement Forum (TMForum) (*TMForum 2007f*). The principal investigation has been a longitudinal Action Research study into the opportunities for open source in the work of the TMForum.

At the start of the program, the TMForum did not have an open source policy or any recognised strategy for engaging with its members on the issues raised by open source. Therefore a more conventional (observational) research approach would most likely have revealed very little of interest. Instead Action Research, which is designed to deliver tangible benefit for all the participants, has been successfully applied with the deliberate objective of forcing the pace of change in the industry and facilitating the emergence of open source contributions.

Research falling within the scope of an Engineering Doctorate program is expected to develop competencies in both the chosen technical discipline and also in R&D, Product Development, Marketing, Program Management, Teamwork and Leadership skills and the ability to apply these skills and knowledge to new and unusual situations (*EPSRC 2006*). The broad scope of the Engineering Doctorate program has allowed the Research Engineer to initiate and drive the projects which delivered the tangible results listed in Table 1. However, more importantly, the research has facilitated a sea change in the attitude in the industry towards open source which potentially will have long-term and profound effects on the development of OSS standards and ultimately OSS solution development.

The program has pioneered the use of Action Research in researching telecommunications OSS, facilitating a multidisciplinary approach to collaboration with industry. The research findings have identified a number of the opportunities, benefits and issues associated with using open source as a rapid innovation methodology in real world industrial settings. The work has provoked considerable industry interest and has led to the academic publications and industry presentations listed in Table 2.

In addition the work has made a significant contribution to developing the potential of the TMForum University Program (*TMForum 2007e*) which seeks to bring academic research to bear on the problems facing the OSS industry. It has lead directly to the industrial sponsorship of another PhD student at the University of Southampton and won a prize for innovation at the 2006 University of Southampton Engineering Doctorate Colloquium.

Table 1 Results from this Action Research Program	
1. The Open OSS Catalyst	
A significant collaborative proof of concept involving;	
Service providers:	BT, Vodafone, Colt, C&W, Covad, QinetiQ, NTT
Vendors:	Agilent Technologies, Sun Microsystems, AutoMagic KB
Integrators:	Cognizant, Invocom
Universities:	Budapest University of Technology and Economics University of Southampton,
2. The TMForum Open Source Program	
As a result of the Open OSS Catalyst, the TMForum instituted an open source program and established an open source repository for future projects (*TMForum 2007d*).	
The intervention also contributed to the launch in 2006 of the TMForum Prosspero Program (*TMForum 2007g*) which aims to solve the problem of slow uptake of paper-based standards by delivering to the development community standards complete with a full support ecosystem to enable rapid implementations[1].	
3. The NGOSSoil feasibility study	
An early deliverable of the TMForum open source program was a feasibility study of an open source project proposed by Nortel Networks.	
4. A DVB-T SLA management proof of concept	
A proof of concept of SLA management for DVB-T networks was conducted using open source components with Arqiva Networks. This study has informed the management architecture for the forthcoming UK digital TV rollout. This study is also a contribution to the convergence of telecoms, broadcasting and content delivery which is a strategic direction for the TMForum.	
Service Provider	Arqiva
Vendors:	Sidonis, OpenNMS
Integrators:	Invocom, Scysis
University:	University of Southampton

Table 2 Papers and presentations	
Papers	
1	Investigating the Feasibility of Open Development of Operations Support Solutions 9th IFIP/IEEE International Symposium on Integrated Network Management (*Gallen, C. R. & Reeve, J. S. 2005b*)
2	The OpenOSS Project - An open Source operations Support Lab for the Telecommunications Industry Workshop on Free/Open Source Software, Technologies and Content (FOSTEC 05), The 9th World Multi-conference on Systems, Cybernetics and Informatics.(IX), Orlando, FA, (*Gallen, C. R. & Reeve, J. S. 2005c*)
3	Using Open Source to Realise an NGOSS Proof of Concept. 2006 IEEE/IFIP Network Operations and Management Symposium NOMS 2006, April 3-7, 2006. (*Gallen, C. R. & Reeve, J. S. 2006*)
Presentations	
4	The future of open source in Operational Support Systems. Conference on Strategic OSS for next generation operators. 18-1-2005. London, IPQC. (*Gallen, C. R. & Pearson, S. 2005*)
5	Using JBoss and the open source approach in the OpenOSS project. JBoss World Telecoms Track Presentation, 10/10/2005, Barcelona (*Gallen, C. R. & Carson, D. 2005a*)

[1] The Influence of this work on the development of Prosspero was independently confirmed by interviews recorded at Telemanagement World May 20-24 2007 with Ian Best, Head of TMForum Product Management and Gary Bruce of BT.

Chapter 1 Introduction: Area of Research Interest

Open source software development relies on gift giving as a way of getting new ideas and prototypes out into circulation.

(Bergquist, M. & Ljungberg, J. 2001)

For anything that Hobbits had no immediate use for, but were unwilling to throw away, they called a manthom... and many of the presents that passed from hand to hand were of that sort.

(Tolkien, J. R. R. 1954 Prologue to The Fellowship of the Ring)

Chapter 1 Introduction to the Area of Research Interest

1.1 Background

An observer of the recent history of the telecommunications industry might be forgiven for wondering if it had fallen under the Chinese curse; 'May you live in interesting times'[2]. In the 1990's an industry boom was driven by the desire to capture market share in the new services made possible by rapidly emerging internet and mobile technologies. The boom hastily gave way to a bust beginning around 1999 when the finance industry started to question the prudence of the service providers' business cases for huge investments in 3G licences and in optical fibre bandwidth - which was projected to far exceed global demand. Confidence was further shattered by the scandal of the WorldCom collapse (*Guardian 2002*). (*Tada, H., Usui, W., & Xu, J. W. 2003*) of NEC referred this as being like a '100-year wave' where an unusual combination of factors produces a wave at sea capable of deluging large vessels. The industry experienced over 500,000 layoffs and €2 Trillion Euros of value wiped out - leaving many suppliers and new operators on the verge of bankruptcy.

Operations Support Systems (OSS) and Business Support Systems (BSS) are the collective names given to software systems which telecommunications service providers rely on to efficiently run their business[3]. (*Tada, H. et al. 2003*) report that 62% of service provider expenses can be directly influenced by an effective OSS and that CFO's are expecting OSS/BSS investments to deliver at least 50% of the OPEX savings needed for the industry to survive. Thus the effective design and deployment of OSS systems is a critical factor in the future profitability of the telecommunications industry.

The research interests of the author are largely driven by having lived through this boom and bust cycle in a large telecoms equipment manufacturer when employed as a Product Manager for their portfolio of OSS products. As part of the Marketing function, a Product Manager's role is to find ways to profitably bridge the gap between developers and end users while keeping both the shareholders and the customers happy. Someone once characterised the impossibility of the Product Managers task by quipping that, 'Product Managers are the only ones who know they are lying'. It is very difficult to maintain a credible product roadmap in the face of a limited development budget and uncertain customer requirements. Thus the

---

[2] The origins of this saying may not in fact be Chinese; see http://en.wikipedia.org/wiki/May_you_live_in_interesting_times

[3] Open source researchers often use the mnemonic OSS to mean Open Source Software however in this Thesis OSS consistently refers to Operational Support Systems (OSS).

Traditionally OSS has referred to network facing functions supporting Fault, Configuration, Accounting, Performance and Security (FCAPS) management of the network. The term Business Support Systems (BSS) has referred to customer facing systems such as customer relationship management (CRM), billing and order management. Traditionally within service provider organisations, OSS systems have fallen under the jurisdiction of Operations departments and BSS system under IT/Finance departments. As hard distinctions between OSS and BSS are disappearing, for simplicity and unless otherwise stated, OSS is used in this Thesis as the collective term given to all of the software components on which telecommunications service providers rely on to efficiently run their business.

researcher wanted to devise a research program which could provide some relevant new insights into the process of delivering OSS solutions; addressing the potential of open source to aid in the product lifecycle of OSS system design, development and deployment in a real business environment.

A key stakeholder in Engineering Doctorate research is the sponsoring company. This program has been sponsored by Invocom (*Invocom 2007*), an OSS systems integrator based in the UK. Early in the program the researcher determined with Invocom that an interpretive research paradigm such as Action Research was potentially closely aligned with their business model and that engaging with the TMForum on the subject of open source could deliver a significant and visible impact for the company. The process by which this research direction was agreed with the sponsor is outlined further in Appendix B.

1.2 Standards and Open Source

Although there appears to be no single universally accepted definition of a standard, the ISO/CENELEC definition provides a useful starting point;

> A Standard is a document, established by consensus and approved by a recognized body that provides, for common and repeated use, rules, guidelines or characteristics for activities or their results, aimed at the achievement of the optimum degree of order in a given context. Standards should be based on consolidated results of science, technology and experience, and aimed at the promotion of optimum community benefits (*CENELEC 2006 Section 3.1*)

(*Murch, R. 2004*) identifies two basic types of standards; Proprietary or de facto ('from the fact'), standards which evolve from a product line or specific vendor and Open or de jure ('by law'), standards which are developed and adopted by some authorised standardisation body. Such Standards Development Organisations (SDO's) may be created by treaty among national governments or as voluntary non-treaty organisations. The TMForum, the focus of this study, is a voluntary trade association which creates Open standards for the OSS industry. Standardisation in the TMForum can be a first step towards standardisation within treaty based standards organisations like the ITU-T. The TMForum eTOM (*TMForum 2007b*) standard has been adopted as ITU-T M.3050 (*ITU-T 2004*) and the TMForum and the DMTF are recognised contributors to the ITU-T NGN Management Focus Group (*ITU-T 2006*).

The standards development process has sometimes been caricatured as a rather uninventive activity by which companies lobby to open up markets for their inventories of technology. Politics and compromise rather than innovation are thought to be the predominant features of the exercise. However, the development of standards in the telecommunications OSS industry actually requires a high degree of collaborative innovation because the solutions being sought are beyond the resources of any one company in the supply chain to develop.

(*Gacek, C. 2004*) has noted that it is not uncommon to see the term open source software being discussed as if it was just another manifestation of collaboratively developed standards. The question of what constitutes an open source project has been the subject of much heated discussion. The Open Source Initiative has set guidelines by which a program may be strictly

considered open source (*OSI 2006*) and (*Laurent, A. M. 2004*) has provided a comprehensive introduction to open source licensing. However in order to avoid debates within the TMForum about licensing and so focus discussion on the potential benefits of collaborative development, a very wide definition of open source coined by the Wall Street Journal was chosen:

> Definition of open source: Open-source programs share programming instructions usually kept secret in commercial software. Such programs are created and improved by global groups of programmers and users. (*Guth, R. A. 2004*)

(*Gacek, C. 2004*) concluded after comparing processes in a generic open source project corresponding to the open source definition (*OSI 2006*) with the IEEE standardisation process that while there were some similarities, open source and standards development projects were in fact distant cousins. The principal differences lay in the suggestion that corporately sponsored experts would be expected to contribute to the standards process while open source projects would be open to anyone and less subject to a formal approval process. However her analysis failed to identify the overlap between standards and open source which has occurred in projects established specifically to provide reference implementations for standards.

As an example, the Java Community Process (JCP) (*JCP 2004a*) involves industry experts in defining new java technology standards. The JCP requires that all Java Specification Requests (JSR's) are backed by a reference implementation (RI) to prove the specification can be implemented, and a technology compatibility kit (TCK), a suite of tests, tools, and documentation that is used to test implementations for compliance with the specification. Although the RI's and TCK's for many JSR's are merely test stubs and not inherently useful (except to developers), some have been turned into full open source products.

The official reference implementation for the JSR053 Java Servlet 2.3 specification (*JCP 2001*) is based on Apache Tomcat (*Apache 2007a*), which is an open source project run under the auspices of the Apache Software Foundation (*Apache 2007f*). The Tomcat project not only provides a reference implementation of the standard but it has created a rugged technology which is widely used both stand-alone and as a component imbedded in commercial products. This suggests that although (*Gacek, C. 2004*) may be right in observing some basic differences between open source and standards development, combining the two processes could benefit both activities. A successful open source project will create a wide user base and will be subject to feedback and continuous improvement. This could be leveraged by a standards creation organisation to ensure the correctness of standards and also to aid with their dissemination and adoption. Conversely an open source project which implements emerging standards will attract early adopting users and potentially more developers.

The focus of this research is to discover the ways in which open source could contribute both to OSS standards creation and adoption and to the general practice of developing OSS systems. The following sections will develop a wider view of the relevance of this approach by briefly

reviewing the history and status of network management practice leading up to the development of NGOSS.

1.3 Brief history of Network Management leading up to NGOSS

(*Pavlou, G., Fegklas, P., Goorveris, S., & Liotta, A. 2004*) provide a brief technical history of network management technologies from early protocol based approaches such as OSI and SNMP through to distributed object based approaches based upon CORBA and latterly on XML. Throughout this technical evolution, the TMForum has performed a unique role as a body which facilitates the standardisation, commercialisation and marketing of these technologies to telecommunications service providers.

The major OSS industry standardisation efforts presently being coordinated by the TMForum are centred around the development of the New Generation Operations Systems and Software (NGOSS) framework (*TMForum 2004b*). Although it is difficult to construct a complete history of the influences leading up to NGOSS (which is merely the latest manifestation of the TMForum's efforts to solve the problem of OSS integration through standardisation), it is informative to consider some of the clearly identifiable milestones.

1.3.1 Early work with TMN

The TMForum was founded in 1988 as the OSI/Network Management Forum with the vision of 'accelerated availability of interoperable network management products' (*TMForum 2007c*). At that time the industry was focused on developing network management solutions based on the ITU-T M3000 Telecommunications Management Network (TMN) standards (*ITU-T 2000b*). TMN partitioned the OSS/BSS space into Element Management, Network Management and Business Management layers. Although it was originally envisaged that network elements could be managed by multi-vendor element managers, the market quickly converged onto a practice whereby element manufacturers created Element Managers which allowed them to differentiate their products and multi-vendor management was pushed up to the Network Management layer.

The TMN standards specified a Q3 interface for sharing information between layers using OSI management standards defined in the ITU-T X.700 (ISO/IEC 7498-4) (*ITU-T 1992a*) series of recommendations. Q3 required software agents which implemented the Common Management Information Service (CMIS) in order to model the network as managed objects defined according to the Guidelines for the Definition of Managed Objects (GDMO) ITU-T X.722 (*ITU-T 1992b*). Communication was achieved using the Common Management Information Protocol (CMIP) (ITU-T X.711) (*ITU-T 1997a*) which ran over the OSI communications stack.

Unfortunately the X.700 series of standards were very complex and contained so many possible implementation options that the probability of successful interoperation between two or more independently developed implementations was quite low without a significant amount

of integration testing. The OSI/Network Management Forum developed a set of recommendations called OMNIPoint[tm] which described a more limited inter-working profile in order to enable faster integration between the Network and Element Management layers.

The complexity and resource requirements of CMIP managers and agents were ultimately there undoing. Today CMIP implementations are mainly confined to legacy systems and to SDH/SONET or ATM element managers where X.700 management has been mandated as part of the transmission standard. Despite the fact that CMIP could be made to run over TCP/IP[4], the IETF's Simple Network Management Protocol (SNMP RFC-1157) (*IETF 1990*) came to dominate the management of TCP/IP devices on the emerging internet. Just as TCP/IP overshadowed the OSI communications stack because of its simplicity and the availability of open source implementations, SNMP required a shallower learning curve and was supported by low cost or open source implementations[5] which significantly reduced the barriers to adoption compared to CMIP based systems.

It should be noted that while the binary based protocols CMIP and SNMP competed for pre-eminence, other ASCII based protocols were (and remain) in common use. Even today most routers are configured using a command line interface (CLI) and many devices report status using a sys-log type interface or TL-1[6]. Although less efficient than binary protocols, the continued popularity of ASCII is partially explained by the fact that human readability makes network management integration and debugging much easier. This also explains the recent interest in XML based management which adds structure to human readable messages and uses commodity e-business technology (*Pras, A., Schonwalder, J., & Festor, O. 2004*). The latest TMForum interface standards MTOSI (*TMForum 2005f*) and OSS/J (*OSS/J 2007b*) both leverage XML for similar reasons.

1.3.2 CORBA and TINA-C Based Management

In the 1990s Telcordia Technologies (then Bellcore) conducted research projects leading to the Operations Systems Computing Architecture OSCA[TM] (*Mills, J. A. & Ruston, L. 1990*) and the Information Networking Architecture INA (*Rubin, H. & Natarajan, N. 1994*). Telcordia's INA work was contributed to the TINA Consortium (TINA-C) (*TINA-C 2007*) which ran between 1993 and 2000[7]. TINA-C used a Distributed Programming Environment (DPE) which was based on OMG's CORBA (*OMG 2007b*) with adaptations for telecommunications requirements (*Ainslie, B. J., Fisher, M. A., & Rana, S. P. 1999*).

---

[4] CMOT - CMip Over Tcp/ip defined in RFC 1095 (*IETF 1989*)

[5] For example NetSNMP http://net-snmp.sourceforge.net/ . This UNIX toolkit for SNMP was first developed in 1992 at Carnegie-Mellon University and made public in 1995 as ucd-snmp v3.0. The project was subsequently used by the IETF in 1998 as the basis for the SNMP V3 reference implementation. It moved to Sourceforge and changed its name to Net-SNMP in 2000

[6] Transaction Language 1 (TL1) is a widely used, ASCII "legacy", management protocol defined in GR-831 by Bellcore (now Telcordia Technologies). See http://en.wikipedia.org/wiki/TL1

Several industrial/academic collaborations are also relevant to the story. The European Union ACTS FlowThrough project AC335 (*FlowThrough 2000a*), (*FlowThrough 2000b*) was a collaborative R&D effort running from March 1998 to February 2000. It sought to demonstrate the implementation of integrated management business processes using reusable components with CORBA providing the business process interaction and backbone distribution services. FlowThrough also contributed research findings to the development of TINA-C.

OSCA™ and INA resulted in a large body of literature dealing comprehensively with the key problems of highly distributed computing which Telcordia contributed to the Telemanagement Forum in 1999. This was condensed and released in 2001 as GB-909, Generic Requirements for Telecommunications Management Building Blocks (*TMForum 2001*) which contained guidelines for OSS product development using CORBA and Java.

During this time a number of OSS products emerged which claimed compliance to the GB-909/TINA-C principles. Although the products worked effectively in their own domains, they never really delivered the levels of integration promised by GB909/TINA-C between OSS systems from different vendors. One example of these was Nortel's Integrated Network Management (INM) technology based on CORBA building blocks which will be relevant to later discussion of Nortel's interest in Open Source in Section 4.3.4.2.

(*Shrewsbury, J. K. 2001*) notes that Telcordia abandoned their INA work in the mid-1990s, largely because the middleware commodities needed to make INA-based products did not emerge from the IT industry in the expected time frame. CORBA had not succeeded in becoming a mainstream IT technology which could be leveraged by the OSS vendors.

Thus by the turn of the Millennium, the OSS industry had made several unsuccessful attempts at using standards to solve the OSS integration problem and there was a growing cynicism in the industry that this could be achieved. Having grown wary of tying its future to standards based on the uncertain success of emerging implementation technologies, the TMForum turned its attention to considering whether the core OSS business logic could be standardised using a technology neutral OSS architecture.

1.3.3 Technology Neutral OSS Component Architectures

The desire to investigate the development of technology neutral OSS architectures and the means of binding of these designs to technology specific implementations motivated a second important industrial/academic collaboration, the EU IST 5th framework project FORM (April 2000 - May 2002) (*FORM 2002*). FORM succeeded FlowThrough with the goal to develop, evaluate, disseminate and exploit an Open Development Framework (ODF) for the emerging market in component-oriented OSS software. According to (*Donnelly, W. & Wade, V. 2001*)

---

[7] The TINA Consortium was formed in 1993 by 46 of the world's leading Telco operators, vendors and IT industry with the mission to define a common software architecture for telecommunications. It was wound up in 2000.

the FORM project was focussed on addressing the interdependencies between the System Development value chain and the Service Provision value chain. A key contribution of the FORM project was the demonstration that an OSS Logical Architecture could be modelled using UML and expressed through a Technology Architecture onto a range of technologies. Demonstrators were implemented using CORBA, EJB and XML technologies. The FORM project is seen as a significant contributor to the early development of NGOSS's concepts of Technology Neutral and Technology Specific architectures.

(*Caruso, F. & Umar, A. 2004*) describe how over a 20 year history of OSS development Telcordia, like the other OSS ISV's, has had to re-implement essentially the same OSS business logic as successive implementation technologies have become fashionable. Prior to the 1990's home grown middleware was used to develop OSS applications. In the early 1990's this was replaced by OSF/DCE (*TheOpenGroup 2007b*). In the late 1990's the OSF/DCE applications were re-engineered to become CORBA compliant. The CORBA-compliant applications were now about to be re-engineered to create loosely coupled publish-subscribe-based applications using use XML based Web Services (*W3C 2004*). Anticipating that Web Services would similarly be replaced at some point, they make the case for 'survivable architectures' which they define as 'architectures which can survive technological and business turbulences'. (*Caruso, F. et al. 2004*) use this story to explain the need for NGOSS as a Technology Neutral Architecture which reduces the pain of constantly re-implementing the same business logic using the latest technology.

1.3.4 TMForum NGOSS

The TMForum's NGOSS has been developed with a view to fostering a market in interchangeable, cost effective OSS components which can be implemented using commodity enterprise IT technologies and which are much simpler to integrate and adapt to changing business requirements than are today's OSS solutions. For a comprehensive introduction to NGOSS see (*Reilly, J. P. & Creaner, M. J. 2005*).

NGOSS is a highly abstract standard which provides a framework for a technology neutral component OSS systems design which can be mapped onto a technology specific implementation. NGOSS consists of a Business Process Framework based upon the enhanced Telecoms Operations Map (eTOM) (*TMForum 2007b*), an Information Framework based on the Shared Information and Data model (SID) modelled in UML (*TMForum 2004f*), an Applications Framework based upon the Telecoms Application Map (TAM) (*TMForum 2006a*) and a Systems Integration Framework based upon a Technology Neutral Architecture (TNA) (*TMForum 2006c*).

The Technology Neutral Architecture is a set of recommendations for specifying, independent of the implementation technology, a telecoms 'Distributed Information Orientated

Architecture' (DIOA) (*Szyperski, C., Gruntz, D., & Murer, S. 2003*). In the NGOSS DIOA, components are described using specifications attached to each interface called Contracts which mutually bind the clients and providers of that interface. Contracts can cover functional and non functional aspects. Functional aspects include the syntax and semantics on the interface. Non functional aspects include the quality of service of the interface. In a fully compliant NGOSS implementation the orchestration of work flow between components (i.e. the business process logic) and the criteria used at decision points (policy) should be maintained in separate work flow managers and policy servers and not be imbedded in each of the OSS components.

In developing NGOSS, the TM Forum has recognised that a pragmatic approach must be adopted in moving the market towards an NGOSS component architecture. Thus while NGOSS looks far beyond currently available OSS systems, it does provide some currently useful tools to help develop solutions using technology available today.

NGOSS provides a methodology and lifecycle for developing a solution from capturing initial business requirements through to deployment using a series of views (Business, System, Implementation and Deployment views) (*TMForum 2004a*). At each stage in the lifecycle, the Contracts, Workflows and Policies are refined with traceability back to the business requirements.

Service providers need a migration path for their legacy systems and software vendors need time to transition existing product offerings towards NGOSS. In support of this, The TM Forum is also developing the Telecoms Application Map (TAM) (*TMForum 2006a*) which offers a taxonomy to describe in NGOSS terms the functional components already embedded within currently purchasable OSS systems. This provides a tool to encourage vendors and service providers to agree a pragmatic mapping between legacy or currently available software packages and a candidate set of NGOSS Contracts which they might be expected to support going forwards.

Although NGOSS has been described as 'a comprehensive integrated framework for developing, procuring and deploying operational and business support systems and software' (*Reilly, J. P. et al. 2005 p12*), the present reality is that while service providers may express a desire for vendors to have an evolution path towards NGOSS, compliance is still rarely mandated in the procurement process[8]. The industry has a fragmented understanding of NGOSS and project managers still seem to fear that requiring NGOSS compliance will add complexity and cost without proven benefits. The previous experience of OSS industry standards outlined above has shown that there is a big cost to backing the wrong horse.

[8] This would be the experience of Invocom who have been involved in numerous service provider RFQ / RFP activities.

Because NGOSS is not complete, the TMForum members are presently encouraged to treat the NGOSS standard as a set of separately usable tools. For instance, service providers have found many uses for the eTOM business process framework beyond that for which it was originally designed (*Milham, D. & Ronco, E. 2004*). The TMForum is seeking to increase NGOSS acceptance through education programs and through 'Catalyst' projects in which OSS vendors are encouraged to demonstrate NGOSS based integration of their products at the TMForum's Telemanagement World (TMW) trade shows. More recently in 2006 the TMForum launched the Prosspero program (*TMForum 2007g*) to provide help for developers. However it can still be argued that NGOSS presently suffers from a lack of public technology demonstrators which can be comprehensively evaluated and trialled in pilot projects by industry stakeholders. Open source implementations of the standard could increase confidence among service provider purchasers to specify standards as part of the procurement process.

1.3.5 TMForum NGOSS and DMTF WBEM

Whereas the TMForum grew up in the telecommunications sector and concentrated upon service providers' interests, a parallel body was founded in 1992 with the initial support of Microsoft and Cisco to meet enterprise management requirements. Originally called the Desk Top Management Task Force, its remit was widened from desktop computing to include enterprise networking management when it was renamed the Distributed Management Task Force (DMTF) (*DMTF 2007a*).

The DMTF has developed a set of standards called Web Based Enterprise Management (WBEM) (*DMTF 2005*), (*Hobbs, C. 2004*), at the core of which as a Common Information Model (CIM) used for modelling network resources (*Bumpus, W. 2000*). A parallel work activity called Directory Enabled Networks (DEN) examined how the CIM model could be mapped into directories that supported the LDAP protocol and an X.500 model (*DMTF 2003*). This offered a technical base for practical policy based network management where, for instance, LDAP directories could be used to hold the dynamic user configuration details for networks providing VPN services.

John Strassner who originally worked on the DMTF CIM and DEN activities later extended this work towards a more comprehensive model called DEN-ng (*Strassner, J. C. 2004b*). Strassner contributed DEN-ng to the TMForum to form the basis of the Shared Information and Data (SID) model (*TMForum 2004f*) which is at the heart of NGOSS and gives it the capability of supporting policy based management.

The NGOSS SID provides a unified information model which describes the relationships between information which can exist at many levels in a telecommunications business. Whereas a data model is concerned with implementation, (*Pras, A. & Schoenwaelder, J. 2003*) state that the main purpose of an information model is to model managed objects at a

conceptual level, independent of any specific implementations or protocols used to transport the data. In theory the SID can aid in the design of OSS systems which translate business concerns (e.g. Service Level Agreements) into policies which are applied as configuration commands at the network level. Conversely it also helps in the design of OSS systems which interpret the business impact of network faults.

Traditionally OSS designs have assumed that the network elements were fairly simple devices requiring orchestration by centralised OSS systems however the demarcation line between the intelligence in the network and in the OSS systems is changing. For instance Motorola is sponsoring research to develop 'Autonomic Networks', in which more of the policy (as described by the SID) is imbedded in the network equipment itself rather than the higher level OSS systems (*Strassner, J. C. & Raymer, D. 2006*) (*Strassner, J. C. 2004a*). The introduction of such systems will require that a holistic and common approach to the OSS information model be adopted across the industry.

For this reason, although the DMTF and NGOSS modelling approaches are somewhat different, the industry is pushing to align the models (*TMForum 2005b*), (*TMForum 2005d*) with the CIM focusing on a resource view of the network and the SID providing an overarching business view. The move to align the models is being driven by equipment manufacturers such as Nortel, Cisco and Motorola who foreseeing the convergence between telecoms and enterprise networks and the potential of autonomic networking, do not want to have to support multiple information models in their element management systems.

A key factor to note is the benefit which the DMTF has derived from open source implementations of its standards. Whereas NGOSS started with the aim of creating a technology neutral framework which encompassed the entire OSS/BSS space and which would ultimately lead to technology specific implementations, WBEM started with the aim of creating and supporting an implementable standard. Thus the DMTF WBEM program has been much more development orientated, supporting implementable specifications with an annual developer's conference and actively encouraging an open source program (see Section 4.3.4.1). Implementation experiences are fed directly back into the standard. This has lead to a number of vendors embedding the WBEM components into their management products; a level of practical adoption which has not yet been seen with NGOSS components.

Therefore, given the common history and the potential convergence between the work of the DMTF and the TMForum, one can argue that a natural next step for the TMForum should be encouraging the development of technology specific implementations of NGOSS components using open source in a similar fashion to the DMTF. These could aid the research into future network technologies, drive the market and also ensure the correctness of the NGOSS standard.

1.4 A Critique of OSS History

Although the aforementioned history is of necessity brief and incomplete, it would be foolish to suggest that lessons have not been learnt from the OSS industry's efforts to solve the OSS integration problem. However the history does beg the question whether there are broader systemic rather than just technical reasons why OSS integration remains such an issue.

Few of the integration technologies (e.g. CMIS/CMIP and CORBA) into which the OSS industry previously poured research dollars have reached mass market adoption. Successful point products have been developed, but the basic problem of low cost repeatable OSS systems integration has not been solved. Furthermore, despite the efforts going into NGOSS, it is still arguably not a standard which a service provider can specify in an RFP and expect a clear response from their OSS vendors.

Since CORBA was the last major technology contender, it is worth considering why it failed despite such heavy industry investment. Henning, who worked on CORBA as a member of the OMG's architecture board between 1995 to 2002 and co-wrote a key text book on CORBA (*Henning, M. & Vinoski, S. 1999*), notes the surprisingly rapid marginalisation of a technology which only a few years before was considered the cutting edge of middleware that promised to revolutionize e-commerce. He provides some interesting insights as to the reasons for the failure of market adoption of CORBA (*Henning, M. 2006*) listed in Table 3.

Table 3 Henning's reasons for the failure of market adoption of CORBA	
Cost:	Broader acceptance of the CORBA platform was limited because early commercial CORBA implementations typically cost several thousand dollars per development seat and runtime royalties for each deployed copy of an application. CORBA never became a commodity middleware technology.
Complexity:	The platform had a steep learning curve and was complex and hard to use correctly, leading to long development times and high defect rates.
Market hype:	Customers believed CORBA was a golden bullet and were requesting its use in RFP's and vendors had no choice but to respond by promising to implement using immature draft specifications.
Technical Flaws:	Fundamental technical flaws persisted in the standard which were not caught because the OMG did not require a reference implementation for a specification to be adopted. This made it possible for a technology that was produced by the world's largest software consortium to suffer from major technical flaws.

Henning recommends that no standard should be approved without a reference implementation as a first line sanity check and that no standard should be adopted without having been used to implement a few projects of realistic complexity. He finally notes that the open source community has done a much better job of adhering to these rules than have industry consortia when delivering workable solutions.

Henning's arguments concern CORBA adoption across the IT industry, but they are equally valid when applied to the Telecoms CORBA based initiatives such as TINA-C. Barriers to adoption such as cost, complexity and standardisation prior to proven implementations were all evident. The early market hype eventually gave way to these fundamentals and the whole experience has lead to a degree of industry cynicism about any OSS standardisation efforts.

Henning's last point about open source is key to this discussion. He posits that the discipline of creating particularly an open source implementation may lead to tangible, deployable standards which are not just paper work exercises but are backed by working technologies which significantly reduce the learning curve for adopters. Furthermore the implementation exercise identifies early any fundamental bugs in the standards before adoption.

(*West, J. 2004*) makes the same point more generally suggesting that while some bemoan incompatibility as a symptom of incomplete specification, the reality is that for complex digital systems standards, the formal specification is inherently incomplete and the actual standard is defined both through the written specification and through actual implementations. Without knowledge of both the formal specification and existing implementations, an implementer faces an extended trial-and-error process in order to discover how other firms have resolved specification ambiguities.

In the same time frame as TINA-C, the very successful open source GNOME desktop project was created which has at its core a CORBA implementation, ORBit (*German, D. M. 2005*) which is used to allow Linux applications to communicate through the desktop. If there are problems with CORBA, they have been overcome by the community and GNOME is delivering value as the strategic desk top for Linux implementations such as RedHat. So one can ask, had there been a strategic commitment to using open source in, for instance, the TINA-C effort, could the outcome have been different?

It is worth noting that, although delivering valuable research results which contributed to TINA-C and NGOSS, the aforementioned (partly publicly funded) projects FlowThrough and FORM did not deliver any lasting implementation code on which future researchers or implementers could build. This also makes it difficult to repeat the results. The only publicly available surviving artefacts which the researcher could find from these exercises are papers, reports and perhaps UML models. One can only assume that over the years other industry players also invested heavily in these technologies before they were no longer considered strategic investments and that any resulting code was left gathering dust on the shelf.

1.5 Perceived Problems with the Standards Development Process

(*Dickerson, K. R. 2004*) outlines a service provider's view (BT) of a number of other problems with the Information and Communication Technology (ICT) standards development process.

Firstly, with increased competition and the economic downturn, operators have been less inclined to cooperate and have had fewer resources and so tended to leave standards development to vendors. Operators needed to re-engage actively with the process.

Secondly vendors have not been good at second guessing operator's requirements. This was felt particularly in the vendor dominated IETF which had produced poor fault management standards (one assumes that he means SNMP) and developed standards such as diffServ which

didn't scale beyond the enterprise. Although the IETF standardisation process (*Bradner, S. 1996*) required two working implementations before standardisation, interoperation did not in itself guarantee that the standards met operators needs.

Thirdly, there were too many ICT standards forums and consortiums for operators to track and influence (as many as 500 - see (*ITU-T 2007*)). Many of these were formed around consortia of vendors who wanted to promote their own technology. Operators needed to find a way as a group to encourage forums to close down or merge by providing strong statements of intent - declaring that only standards from their preferred forums would be acceptable and used in procurement.

Given these complaints, it is valid to ask if open source could provide a mechanism which would compliment the standardisation process. Firstly, whereas the payback from reviewing draft paper standards is somewhat deferred, working implementations might make it much more attractive for practitioners within the service providers' organisations to provide useful end user feedback. This might provide a cost effective means to encourage service providers to get involved with vendors in standards development by using reference implementations in realistic scenarios. Secondly, service providers could choose an appropriate level of participation - looking at the usefulness of the technology and not at the details of its implementation. Thirdly, service provider participation in an open source project sends a strong message of their intent to use the standardised technology at some point in the future.

Finally (*Dickerson, K. R. 2004*) blames what he terms an 'IPR morass' for significantly slowing standards implementation. He describes the IPR policies adopted by Standards Development Organisations (SDO's) as falling under two main categories; royalty free (RF) and fair, reasonable, and non-discriminatory (FRAND). The Telemanagement Forum currently uses a version of the FRAND model (*TMForum 1989*). Under the FRAND model a 'timely' declaration of any IPR thought to be essential to a standard under development must be made so that potential users can consider likely licensing royalties. Problems can arise with the FRAND model when vendors disclose their patentable interests in a contribution late in the standardisation process or when vendors disagree as to what is 'fair and reasonable'.

(*Dickerson, K. R. 2004*) notes that there is a suggestion that that traditional standards bodies no longer work due to a complex mass of claims and counterclaims for IPR that are considered essential to a standard's implementation and that the pressure to bring technology to market is leading to a developing tension between two camps in the industry. On the one hand some companies that potentially have significant IPR in standards under development are refusing to join or withdrawing from membership of these bodies, often resulting in the establishment of de facto or proprietary standards (sustained by market power) instead of more open standards. On the other hand there is a 'trend towards openness', illustrated by the members of SDO's moving towards the RF model. New forums such as the OSGi (*OSGi 2007*) and Liberty

Alliance (*LibertyAlliance 2007*) are adopting an RF policy and older forums such as the W3C (*W3C 2007*) and IETF (*IETF 2007*), are moving in that direction. (*Dickerson, K. R. 2004*) believes that ultimately standards bodies will only be successful if they exhibit a flexible policy which encourages RF but allows for FRAND in exceptions. Since the open source model can clearly benefit from and contribute to standards created under the RF model, the thoughtful application of open source may represent a key strategy for SDO's to maintain their market position.

1.6 Perceived Gaps in Network Management Research

Taking this argument further, it is worth considering the nature of Network Management[9] research. Although perhaps not an rigorous survey, it is informative to review some of the statements made by the senior researchers representing vendors, service providers and academia who attended the 2006 IRTF Network Management Research Group (NMRG) Workshop on the Future Direction of Network and Service Management Research (*NMRG 2006a*), (*NMRG 2006b*). While a significant number of detailed technical research questions were elaborated, key concerns were also expressed particularly by the service providers and vendors concerning network management research as a discipline.

Firstly there appeared to be a lack of clear definition around the field of network management and a lack of accepted and explicitly stated research methodologies. There was also a lack of repeatability. Olivier Festor, scientific leader of the joint INRIA-Nancy Universités MADYNES research team stated that an analysis of some 100 papers revealed that only about three papers provide the information necessary to redo the experiment.

The researcher would add that most TMForum Catalyst experiments and to a certain extent the FORM and FlowThrough projects discussed in Section 1.4 could also not be repeated because of the difficulty in assembling the required software artefacts.

Secondly much research was not relevant to end users. Prof Petre Dini, principal architect at Cisco Systems, lamented the 'disconnect between research communities and engineering communities' and noted the 'trade-off between research that is nice to do and the research that has at the end some practical value'. Prof Aiko Pras of the University of Twents stated that it was essential 'that researchers start from real problems, as identified by real managers, and not from artificial problems that look well on paper but have little relationship with reality'. It was stated that PhD students often had no clue how real networks operated and needed to talk more to operators.

Furthermore researchers seemed to prefer the hype of new technology rather than address the practical problems of existing implementations. Often researchers came with 'solutions looking for a problem rather than addressing a problem where we do not really know what is

the right solution?' Questions were also asked as to why practitioners were not using the technologies already developed and why new technologies were not being adopted. For instance why had the work on the IETF Next Generation Structure of Management Information (SMIng) (*NMRG 2003*) not been generally adopted?

Finally the 'boundaries between the technical and business impacts of network management' were poorly addressed. Claudio Bartolini of HP stated the need for understanding of the business relationship of management; how do IT management processes best support business objectives? Others stated the need to look explicitly at the OPEX benefits of network management technologies. In a breakout session, service providers indicated that research was required into the integration of network and service management with business processes and goals but that it was unclear that the network management community was qualified to do this as it wasn't core computer science work.

Thus there appeared to be evidence of some gaps in the way academic network management research was currently being conducted. The researcher wanted to investigate firstly if open source could address some of the issues concerning the repeatability of experiments and the dissemination and practical application of work. Secondly, the researcher wanted to investigate whether Action Research could address the issues of active participation with industry and combine a holistic view of business and technology concerns.

1.7 The TMForum's contribution to industry knowledge

It is worth further noting that there are a whole body of conferences, papers and activities which are not considered 'academic' but which significantly influence the practical direction OSS industry.

As an industry body providing a focal point for the commercialisation of emerging OSS technologies, the TMForum runs a number of working groups. These are staffed by technical specialists drawn from the member companies who work on standards or activities of common interest and publish recommendations or reports. The TMForum runs two Team Action Weeks a year to facilitate these teams meeting together.

The TMForum also runs two annual Telemanagement World (TMW) industry conferences and trade shows. TMW papers are chosen by the conference committee but are mostly self validated and often take the form of case studies where a company tells the industry how well a particular project went in their setting.

This contrasts with, academic research reported at learned society conferences such as the IEEE/IFIP Network Operations and Management Symposium (NOMS) (*IEEE/IFIP 2007*) or the IFIP/IEEE Symposium on Integrated Management (*IFIP/IEEE 2007*). Academic papers are subject

[9] In this context Network Management is taken to refer to the essential function of OSS systems.

to a stringent blind review process which goes far beyond that used in TMW (although, to be fair these conferences are now allowing 'experience papers' which are less rigorous).

TMW's are also characterised by 'Catalyst' projects which are demonstrations of technology developed in 6 monthly intervals by collaborative teams drawn from across the industry. Catalysts may result in academic publication (e.g. (*Caruso, F. et al. 2004*)) but this is not their primary audience. A number of the benefits thought to arise from Catalysts (*TMForum 2007a*) are listed in Table 4.

Table 4 Perceived Benefits of TMForum Catalyst Program
1. Catalysts are thought to accelerate industry adoption of TMForum standards by providing practical demonstrations of the work to the wider industry.
2. The TMForum standards are thought to be strengthened through feedback obtained through Catalyst activities.
3. Organisations participating benefit through the learning exercise facilitated by working on a common project across inter company boundaries.
4. They provide both inbound and outbound marketing opportunities for ISV's wishing to interact with potential service provider customers.

The important question to be considered is; how does one assess the validity and value of the relative truth claims of presentations and catalysts presented at a TMW event over and against the scientific papers addressing similar subjects presented through a learned society like the IEEE? There appears to be a dichotomy in many industry minds which suggests that 'proper research' is pre-competitive, concerned only with the facts and not influenced by commercial considerations. TMW however is considered to be a marketing event where any truth claims should be treated with caution; 'caveat emptor' (let the buyer beware). Yet there must be some value in what is presented at TMW, otherwise important decision makers would not bother to attend.

1.8 Summarising the Research Theme

Section 1.4 surveyed the various failed attempts within the OSS industry to solve the problem of OSS integration. It was suggested that there might be systemic reasons why the OSS industry found this such a difficult problem to solve and that an open source approach might make a contribution towards solving the problem.

It can be argued that the activities of the TMForum discussed in Section 1.7 seek to fill a gap which the more 'academic' network management research is not presently equipped to address; i.e. 'the boundaries between the technical and business impacts of network management' referred to in Section 1.6.

Furthermore much of the work being done academically, reported as case studies by industry or developed through TMForum Catalysts is not easily repeatable and therefore difficult to evaluate. Open source could be used more methodically to assist in the validation and dissemination of TMForum standards in a more public and repeatable manner as discussed in Section 1.5

Thus there is a need for these two worlds of OSS industry and research to be brought together such that they compliment each other in a more rigorous fashion. This points to a requirement to articulate a research methodology which can be validated and recognised as (publishable) academic research while at the same time bridging the gap between the business, research and engineering communities and making the work more relevant to end users.

This leads to the research theme which is in two parts. The primary question concerns,

> 'whether the considered and strategic adoption of open source could further the practice of Telecommunications Operations Support?'

The secondary question which underpins this enquiry asks,

> 'whether one can utilise a research methodology which in a more rigorous way bridges the gap between end users, integrators, developers and researchers in the Network Management field?'

1.9 Relevance of results

It is hoped that the results from this work will be relevant in a number of ways;

Firstly it is hoped that it will offer an approach to OSS research which can make it more relevant to industry while retaining its academic credibility. Hopefully this will encourage more multidisciplinary research to be undertaken in the OSS field.

Secondly it is hoped that the results of this research will particularly help Product and Development managers in the OSS industry to consider more clearly if and where open source could contribute to solution development.

Thirdly it is hoped that it will offer a model which could be used to enhance collaborative solution and standards development efforts in other industries.

1.10 Organisation of the Thesis

As has been alluded earlier the research paradigm underlying this work is Participant Action Research which will be discussed in Chapter 2. However the choice of Action Research raises certain challenges in presenting the work. (*Avison, D., Lau, F., Myers, M., & Neilsen, P. A. 1999*) note that often IS Action Researchers have large and complicated stories to tell such that much of it is published in books rather than in articles. (*Checkland, P. & Howell, S. 1998*) go further and state that in order to distinguish an Action Research account from novel writing, it is essential to first define the research epistemology (i.e. the Theoretical Framework and Methodology) in terms of which what will count as knowledge from the research will be expressed.

Noting these issues, this Thesis is laid out as in Table 5, Following the guidance as to the structure of an Action Research thesis offered by (*Dick, B. 1993*) & (*Dick, B. 1997*), the results will be presented as a dialectic between the literature and reflections on the experience of interacting with the client-systems represented in each project.

Table 5 Organisation of this Thesis	
Chapter 1	Introduction to the Area of Research Interest The introduction (this chapter) has outlined the OSS problem space and broad research themes in terms of their expected contribution to filling a gap in current OSS practices.
Chapter 2	Research Methodology: Action Research The research methodology will review the IS Action Research methodological literature and justify the overall approach to using IS Action Research in the Client-Systems represented by each project. It will also outline the chronology of the research.
Chapter 3	The OSS Industry and Open Source Problem Space This chapter will present a theoretical framework and market analysis through which the potential of open source in the OSS industry will be articulated. This represents the research epistemology (i.e. the Theoretical Framework and Problem Solving Methodology) underlying the following interventions. The following chapters will each be constructed around the three separate interventions which were undertaken during the research
Chapter 4	Intervention: OpenOSS in the TMForum A discussion of the OpenOSS Catalyst project and the follow on NGOSSoil attempts to launch a large open source project within the TMForum by a large equipment vendor.
Chapter 5	Intervention: Digital TV Service Management A collaborative project to apply Open Source to solving a real digital television (DVB-T) SLA management problem.
Chapter 6	Intervention: Contributing an OSS/J Interface to OpenNMS An individual contribution to an existing Open Source Network Management Project run by a small OSS ISV, (OpenNMS).
Chapter 7	Discussion and Conclusion Although lessons will be drawn out in the separate evaluations of the interventions in Chapters 4, 5 and 6, this chapter will summarise lessons learnt from the overall program.

Chapter 2 Research Methodology: Action Research

Chapter 2 Research Methodology: Action Research

This chapter will justify the choice of Action Research as the underlying research methodology and describe how it was applied in each of the industrial interventions. Appendix B provides additional background as to the commercial reasons for choosing to use an Action Research methodology with the sponsoring company.

2.1 Defining Information Systems Action Research

Although relatively recently applied to Information Systems (IS), (*Howell, S. 2004*) notes that Action Research has had a growing presence in the field evidenced by the publication of IS journal issues dedicated to Action Research ((*Information Technology & People 2001*) & (*MIS Quarterly 2004*)). Additionally a recently reader has collated the key IS Action Research papers published over the last 10 (*Kock, N. F. 2007a*).

As Action Research has matured it has acquired numerous definitions but is perhaps best understood not as a methodology or set of techniques, but as

> 'an orientation to inquiry which seeks to re-integrate the dualisms that have haunted Western approaches to knowledge which have hither to split apart theory and practice, researcher and subject, everyday experience and academic knowledge' (*McArdle, K. L. & Reason, P. 2007*).

(*Avison, D. et al. 1999*) suggest that Information Systems and Software Engineering can benefit from Action Research methods because their application domains include the ill-structured, fuzzy world of complex organisations which are not addressed by conventional information systems development methodologies.

According to (*Susman, G. I. & Evered, R. D. 1978*) and (*Kock, N. F. 2007b*) the origins of Action Research lie with psychologists in the 1940's including Kirt Lewin's work on researching social processes in the USA and the work of Tavistock Institute treating World War II battle fatigue patients in London. (*Blum, F. H. 1955*) simply defined Action Research as the 'diagnosing of a social problem with a view of helping improve the situation'. He defined this in two stages; i) the Diagnostic stage in which a problem is being analysed and hypothesis are developed; ii) the Therapeutic stage in which hypothesis are being tested by a consciously directed change experiment preferably in a social life situation.

(*Rapoport, R. N. 1970 p 499*) defined the aims of Action Research as endeavouring '...to contribute both to the practical concerns of people in an immediate problematic situation and to the goals of social science by joint collaboration in a mutually acceptable ethical framework'. (*Susman, G. I. et al. 1978*) suggested that that Action Research should also aim to 'develop the self help competencies of people facing problems'.

The model of Action Research described by (*Susman, G. I. et al. 1978*) is one of the earliest to be applied to IS and is referred to as Canonical Action Research in IS by (*Davison, R., Matinsons, M. G., & Kock, N. F. 2004*). Susman considered that an Action Research activity must be defined not just in terms of its aims but also by a definition of the Client-System and

the Process. The Client-System (or research environment) was the social system (for example a face to face group, an organisation, a network of organisations or a community) in which the members faced problems to be solved by Action Research. The Action Research process was defined by the five phase, cyclical process listed in Table 6.

Table 6 Action Research Activities (after Susman, G. I. et al. 1978)	
(1) Diagnosing	Identifying a problem
(2) Action Planning	Considering alternative courses of action for solving a problem
(3) Action Taking	Selecting a course of action
(4) Evaluating	Studying the consequences of an action
(5) Specifying Learning	Identifying general learning's

Although this gave a comprehensive definition of Action Research, (*Susman, G. I. et al. 1978*) noted that specific Action Research projects could differ in the number of phases which were carried out and the extent to which the 'infrastructure within the client system' (i.e. the participating client's organisation) and the action researcher jointly maintained and regulated these phases. Drawing on (*Chein, I., Cook, S. W., & Harding, J. 1948*), they summarised the options using the four categories of Action Research listed in Table 7.

Table 7 Categories of Action Research (after Susman, G. I. et al. 1978)	
Diagnostic Action Research;	When the researcher only collects data for diagnosis and feeds back to the client system.
Empirical Action Research;	When the researcher only evaluates the actions taken by the client system and feeds this data back to it.
Participant Action Research;	When diagnosing and action planning are carried out in collaboration between the researcher and the client system.
Experimental Action Research;	When the researcher and client systems participate in all or nearly all phases to set up an experiment for taking an action and evaluating its consequences.

2.1.1 Action Research Epistemology

(*Checkland, P. et al. 1998*) note that it is common to use different criteria for judging the truth of discoveries made using different techniques of knowledge acquisition. Quantitative research is more easy to defend because it can be validated by the repeatability of the experiments. Action researchers cannot repeat their experiments and so they need to carefully construct their research methodology to ensure that they do more than just report a 'plausible' research story.

Checkland's model of Action Research knowledge acquisition is illustrated in Figure 1 and states that any research can be characterised by a Framework of ideas (F), embodied in a Methodology (M) which is applied to an Area of concern (A). F and M represent the Epistemology of the research (i.e. the set of ideas and the methodical process in which they are used) and also provide the terms in which the learning's from the research will be expressed.

(*Checkland, P. et al. 1998*) propose that a key criterion for Action Research is Recoverability which corresponds to Repeatability in Quantitative research. Although the unique situation of the Action Research cannot be replicated, enough information must be provided to allow anyone interested in subjecting the research to critical scrutiny to understand how the researcher used the data and methodology to reach their conclusions and acquire new

knowledge. This requires an explicit statement of the Epistemology of the research at each stage of the Action Research cycle.

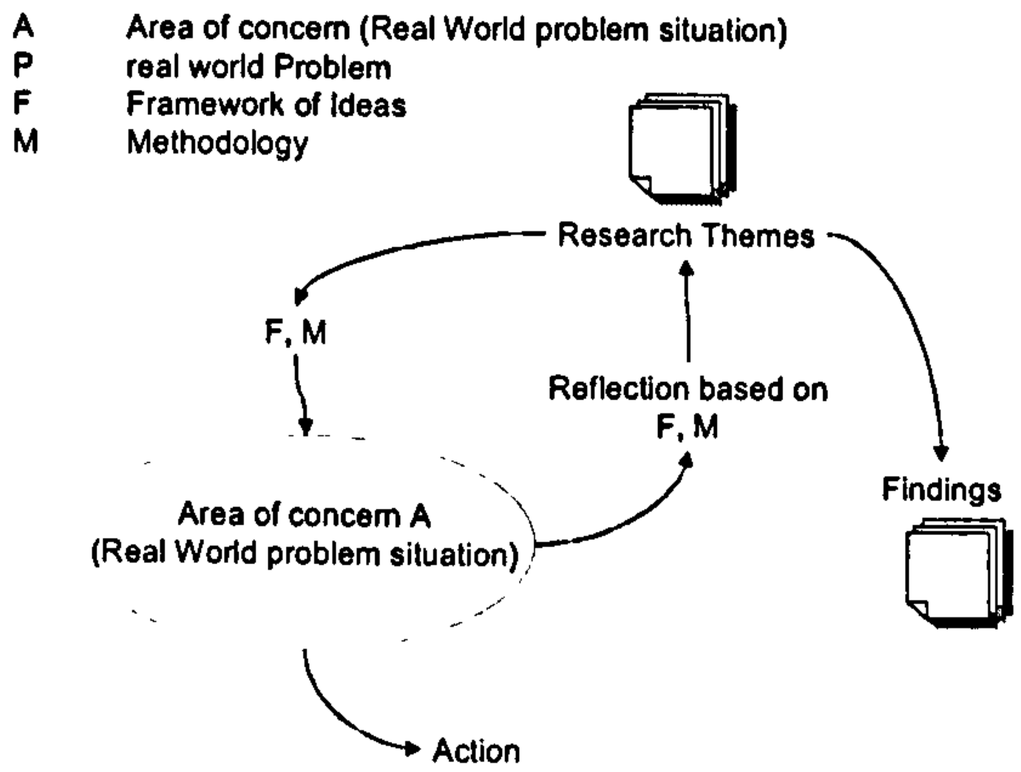


Figure 1 Action Research cycle
(after Checkland, P. et al. 1998 as drawn in McKay, J. & Marshal, P. 2007)

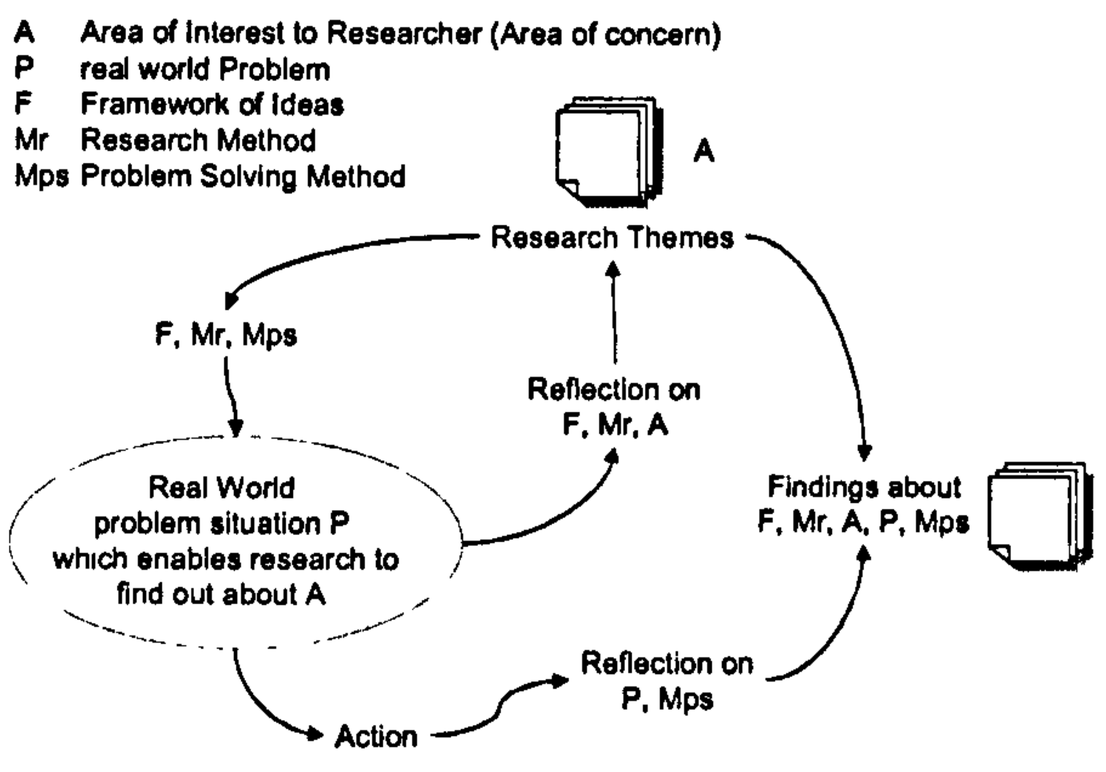


Figure 2 Revised Action Research Framework
(after McKay, J. & Marshal, P. 2007)

(*McKay, J. & Marshal, P. 2007*) argue that it is necessary to extend Checkland's model in order to explicitly and separately address both the problem solving interest and the research interest of the action researcher. They suggest that Action Research should then be considered as two interlocking cycles of interest; the iterative process of applying a methodology to solve a specific problem in a client system and the (separate) iterative process of using Action Research to generate new knowledge. This is illustrated in Figure 2. Separating reflection on the research methodology (Mr) from reflection on the problem solving methodology (Mps) allows the research methodology to be examined and iteratively improved as a separate activity to reflecting on the learning's about the problem solving methodology itself.

(*McKay, J. et al. 2007*) make another subtle improvement by suggesting that it is useful to separate the declaration of the Area of interest to the researcher (A) as expressed in the research themes, from the real world Problem (P) which has been identified to address some aspect of researcher's area of interest. To assist the prospective action researcher, guidance is also given by (*McKay, J. & Marshal, P. 2002*) to help develop an Action Research project.

2.1.2 Action Research Process

Figure 3 illustrates Checkland's version of the Action Research process (*Checkland, P. et al. 1998*) which can be considered as complementary to (*Susman, G. I. et al. 1978*) and (*McKay, J. et al. 2007*) described above. The following sections will consider in more detail each step of the process.

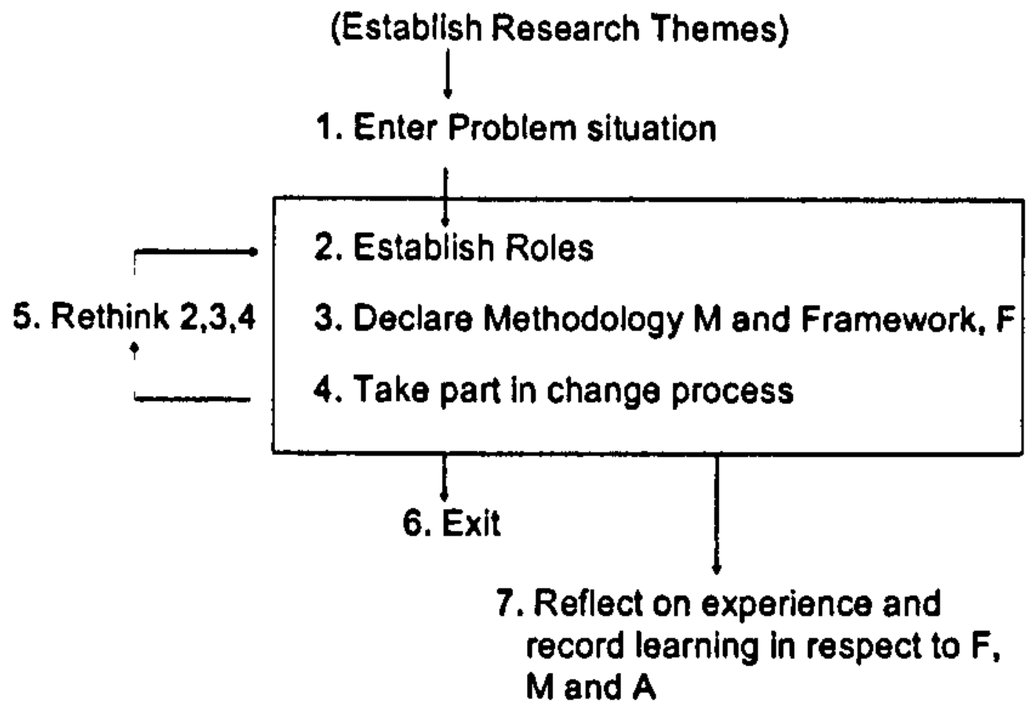


Figure 3 Action Research Process
(after Checkland, P. et al. 1998)

2.1.3 Research Themes

Whereas a Positivist researcher initiates their research by seeking to prove a *hypothesis*, an Action Researcher states a set of *research themes* within which lessons can be sought, seeks a real-world situation in which the themes are relevant and becomes involved as both participant and researcher.

(*Howell, S. 2004*) notes that research themes can be a longer term, broader set of questions that motivate the researcher and go beyond a particular context. Research themes are unlikely to be completely resolved through a single intervention and the linking of projects via research themes means that iteration can be thought of as more than just the iteration around one Action Research cycle. This allows more than one project to address aspects of a particular research theme through a research program.

2.1.4 Entering the Problem Situation

Having observed that previous Action Research literature had tended to gloss over the problem of gaining access to relevant problem situations, (*Kock, N. F., McQueen, R. J., Baker, M., & Rouse, A. 1996*) propose a model for negotiation and note that the success of such search and negotiation activities is a major risk to Action Research projects. They note that the process involved in matching the research objectives to a client's needs can be very time consuming and severely tests the negotiation skills of the IS researcher.

Having obtained access, (*Mumford, E. 2001*) recommends the creation of a formal 'action document' with a precise specification of processes, objectives and outputs so that all contacts in the organisation have an unambiguous, specific and agreed knowledge of what is to take place. She recommends that the researcher continuously make an effort to keep explicit communication channels open to avoid any unpleasant surprises as the project proceeds.

(*Avison, D., Baskerville, R., & Myers, M. 2007*) consider the issue of power to be an often neglected but very important aspect of an Action Research project. They identify four sources of power; Authority - derived from formal or institutional structures; Resource Rights - derived from ownership over resources; Influence - a source of power derived from trust or charisma;

Politics - a source of power derived from exercise of strategic processes. Understanding the motivation and power relationships between players in a situation can be very important to the success of a project.

2.1.5 Establishing Roles

Since the researcher is also a participant in the process, (*Checkland, P. et al. 1998*) note that it is necessary to think carefully about the dual role and negotiate carefully entry into the situation. Furthermore the very act of seeking access must be considered as part of the Action Research process since it itself represents a significant intervention (*Coughlan, P. & Coghlan, D. 2002*).

Action Research has attracted the criticism that it is almost synonymous with consulting and it is important to make the differentiation in the clients mind. (*Mumford, E. 2001*) states that it is important for Action Researchers to protect their 'academic neutrality' and distinguish their role from that of the professional consultant, stating that while some researchers act like consultants it is rare for consultants to act like researchers and make a major contribution to theory. (*Baskerville, R. 1999*) outlines 5 key differences between Action Research and Consulting in terms of Motivation (scientific vs. commercial), Commitment (to the scientific community and client vs. only to the client), Approach (collaboration vs. 'outsiders', unbiased viewpoint), Foundation for recommendations (theoretical framework vs. experience) and Basis of Organisational Analysis (iterative experimental changes vs. independent critical analysis).

(*Mumford, E. 2001*) also identifies potential ethical issues in that the consultant is paid to further the interests of a particular group, usually the management. The academic researcher, in contrast is, ideally, dedicated to the pursuit of knowledge in an ethical manner and is there to assist, not damage, the interests of the group or groups that are being studied.

However (*Davison, R. & Martinsons, M. G. 2007*) respond to this criticism by stating that there are considerable opportunities for Consulting and Action Research to leverage their respective strengths. They cite cases where the organisational knowledge management within consulting practices could be greatly improved through the application of Action Research principles such as those they articulated for Canonical Action Research (*Davison, R. et al. 2004*).

2.1.6 Reflective Learning

(*Schön, D. A. 1983*) first introduced the idea of the professional as a 'Reflective Practitioner'. To improve, professionals must not only learn more theory, they must also think about what they have done, an activity which he described this as 'reflective practice'.

Reflective learning is often cited as a key practice for action researchers who must be able to reflect on the nature and impact of their own actions. (*McArdle, K. L. et al. 2007*) note that reflection can happen 'on-line', as the project proceeds or 'off-line', 'after the event'. On-line

reflection can be very challenging as it demands multi level attention whilst acting; addressing both the problem in hand and reflecting upon the approach.

It is important therefore to use reflective practice to consider the effect of participating in an open source activity. This approach has been used elsewhere for developing the practice of software engineering and in particular for extreme programming (*Hazzan, O. & Tomayko, J. E. 2005*), (*Hazzan, O. & Tomayko, J. E. 2003*), (*Hazzan, O. 2002*)

2.1.7 Overall Learning Cycle

It is also necessary to consider the overall learning to be acquired throughout the project. One of the complaints about academic network management research previously noted in Chapter 1 (Section 1.6) was that the researchers were not sufficiently knowledgeable about the real operations environment. (*Coughlan, P. et al. 2002*) define pre-understanding as the knowledge the researcher brings to the project and state that Action Researchers in operations management (an area closely aligned with Network Management) would need to have not only their knowledge of operations and production but also a broader understanding of organisational systems.

To be effective (and to participate equally with the client system) an action researcher requires a breadth of pre-understanding of the corporate environment, the conditions of business, the structure and dynamics of operating systems and the theoretical underpinnings of such systems. Such knowledge is not acquired over night.

Furthermore, an Action Researcher also requires a level of capability in Action Research. (*Estay-Niculcar, C. A. & Pastor-Collado, J. A. 2002*) have defined a helpfully maturity model Table 8, for project managing an information systems action-research project. This mirrors the Software Engineering Institute's Capability Maturity Model (*SEI 2000*).

Table 8 Project Management Maturity Model For an IS action-research project
(after Estay-Niculcar, C. A. & Pastor-Collado, J. A. 2002)

Level 1 Novice	This level is characterised by a general knowledge, principally literary, of Action-Research, reflected in the reproduction of actions. The success of IS-AR depends mainly on the innate characteristics of the novice researchers in what they understand for IS-AR, on the facilities contributed by the practitioners, and on luck.
Level 2 Basic	The success of the process is obtained by following basic criteria that allows justifying the use of Action-Research.
Level 3 Organised	Project management practices are fully introduced
Level 4 Managed	Risk and quality project management processes are added with profusion.
Level 5 Adaptive	Expert, continuous, sometimes automated, creative and sustainable use of the results and experience are accumulated. Thus the action researcher evolves, learns and adapts her/his experience through learning and conversations with other researchers and practitioners.

2.1.8 Quality in Action Research

Action Research has been criticised as an approach in IS because of the perceived lack of methodological rigour in the reporting of many projects ((*Lau, F. 1997*), (*Lau, F. 1999*), (*Avison, D. et al. 1999*) and (*Davison, R. et al. 2004*)). However (*Howell, S. 2004*) considers

that the earlier debates about the validity of using qualitative techniques in IS have been superseded by a more recent discussion of what constitutes good practice.

In seeking to establish a framework for rigour in Qualitative Action Research, (*McKay, J. & Marshal, P. 2000*) drawing on (*Guba, E. G. & Lincoln, Y. S. 1989*), propose design tests which correspond to quality criteria often applied to Quantitative research. The criteria are summarised in Table 9. ((*Riege, A. M. 2003*) has undertaken a similar exercise for qualitative Case Studies).

Table 9 A framework for rigour in Qualitative Action Research
(after McKay, J. & Marshal, P. 2000)

Quantitative	Qualitative
Internal validity Internal validity, refers to the establishment of cause-and-effect relationships	Credibility • Refers to the way in which the complexities of the research setting are presented. • Requires that the parameters of the research are adequately stated (in terms of setting, participants, theoretical framework and method). • Research is credible when it presents such faithful descriptions or interpretations of a human experience that the people having that experience would immediately recognize it from those descriptions or interpretations as their own. • A study is also credible when other people (other researchers or readers) can recognize the experience when confronted with it after having only read about it in a study.
External validity External validity is concerned with the extrapolation of particular research findings beyond the immediate form of inquiry to the general.	Transferability • Others can decide whether the results of an action research study apply to other settings if the researcher uses an explicit theoretical framework to explain and interpret the outcomes of actions in a problem situation. • The onus of transferability rests more with the transferor, rather than with the original researcher. • Confidence in transferability of findings is likely to increase the more iterations of the action research process.
Reliability Reliability refers to the demonstration that the operations and procedures of the research inquiry can be repeated by other researchers which then achieve similar findings	Dependability • Relies on a transparent process that is established, traceable and documentable and hence open to scrutiny. • Depends on the auditability of the research process or the ability of another researcher to follow the "decision trail" of the original researcher.
Objectivity Objectivity assesses whether the interpretation of data is drawn in a logical and unprejudiced manner	Confirmability • Requires that research data can be traced back to their source, and judgements and assertion made about that data are logical and coherent, and are able to be confirmed by an expert other than the researcher. • The concern about possible researcher bias and subjectivity are removed if the findings can be demonstrated to grow out of the data.

2.2 Research Method used in this program

Having broadly discussed Action Research as a methodology (Mr), the following sections will consider more specifically how Action Research was employed throughout this longitudinal study with members of the TMForum. The research method has drawn on the variety of approaches to Action Research described above.

2.2.1 Participant Action Research

Participant Action Research as defined in Table 7 by (*Susman, G. I. et al. 1978*) (also known in IS as Canonical Action Research (*Davison, R. et al. 2004*)) should be considered as the over arching research paradigm adopted in this program. Diagnosing, Action Planning and Action

Taking were carried out in collaboration between the researcher and all the project participants but the detailed analysis (i.e. Evaluating, Specifying Learning) has been done exclusively by the researcher.

In practice however it was unrealistic to try and impose a rigid Action Research cycle on the natural occurrence of business events in each project (project meetings, presentations etc.). Instead it has been helpful to use the 5 Action Research stages described by (*Susman, G. I. et al. 1978*) in Table 6 to characterise the contribution of each of the key business events. This approach has been used to give structure to the chronology of the program described in Appendix A.1.

The research has been conceived around the less prescriptive Action Research cycle proposed by (*McKay, J. et al. 2007*) (Figure 2) which extends that of (*Checkland, P. et al. 1998*) (Figure 1). In this formulation, the Area of Interest for the researcher (A) has already been described by the research themes introduced in Chapter 1. The remainder of this Chapter 2 will describe specifically how the Action Research methodology (Mr) has been applied in this program. In Chapter 3 the real world problem situation will be described in terms which industry players would recognise (a key quality criteria from Table 9 (*McKay, J. et al. 2000*)). This summarises a diagnoses of the problem (P) which the industry is seeking to address through bodies such as the TMForum. It also provides a Framework of ideas (F) which theorise how the collaborative innovation fostered in the standards organisation combined with open source might be applied as a Problem Solving Methodology (Mps). Thus Chapters 1-3 set out the overall research epistemology. The research will be presented in terms of a dialectic between the theory (epistemology) and the findings of a set of projects.

The research has been conducted as a number of essentially separate projects (Interventions) linked through the common research themes as suggested by (*Howell, S. 2004*). This approach was necessary for pragmatic reasons since it has not been possible to assemble the same parties to go through more than one iteration of each intervention.

Each intervention was created to addresses a particular region of the problem space. A broad mapping of the projects onto the problem space is explained in the research program design in Section 3.7. Further clarification is provided in the diagnosis section of each intervention description. The project has demonstrated a broad coverage of the issues raised in Chapters 1-3 but not every issue raised by the problem space has been addressed by the projects. This leaves room for future research.

Finally, the Participant Action Research methodology (Mr) has been treated as an over-arching 'orientation to inquiry' (*McArdle, K. L. et al. 2007*) guiding the use of the other tools discussed in Appendix B.3 (Table 42) in Appendix B. For instance, the OpenOSS Catalyst (Section 4.2) and the DVB-T digital TV (Chapter 5) projects both represent Concept Implementations (or

proofs of concept) while NGOSSoil (Section 4.3) is more of a Case Study since nothing was actually developed. The work done with OpenNMS discussed in Chapter 6 could be considered an exercise in reflective learning.

2.2.2 Learning Cycle

The entire program has been characterised by reflective learning in a learning cycle as discussed in Sections 2.1.6 and 2.1.7. One of the main purposes of the OpenOSS catalyst was for the participants to investigate the potential for open source in OSS through collaborative learning. Each participant had the opportunity to take away their own learning's from their involvement. For instance the BT participants leveraged the OpenOSS catalyst as one aspect of a research activity which was being conducted within BT and which ultimately informed both a BT research paper (*Bruce, G. L., Wittgreffe, J. P., Potter, J. M., & Robson, P. 2005a*) and a Eurescom study (*Bruce, G. L., Wolland, S., Wiener, J., Silva, N., & Ryan, A. 2005b*). In the case of the undergraduate student participants, the educational value of the exercise was an explicitly stated objective as part of their degree programs.

While agreeing with (*Coughlan, P. et al. 2002*) that an action researcher's pre-understanding of the industry is very important, one would add that the experience of this project suggests that the combined skills of a participant Action Research team can make up for the researchers lack of knowledge in certain areas. For instance, the feasibility of the OpenOSS Catalyst depended upon the level of capability within the team in terms of prior knowledge of the industry context, network management, software development, business practice and project management. In the case of the researcher (Craig Gallen), his industry background meant that he had a prior understanding of the industry context from an engineering and Product Management perspective. However in the Catalyst project he and the undergraduate students had to learn from scratch how to use open source tools and how to program J2EE applications. Others in the team had stronger programming skills and were able to bring the rest of the team up to a satisfactory level much quicker than if they had been left to their own devices.

The same observation should be made with respect to the researcher's early knowledge of Open Source and of Action Research methodology. Thus the statement made by (*Checkland, P. et al. 1998*) that research recoverability requires an explicit statement of the Epistemology of the research (in terms of the framework of knowledge (F), the research methodology (Mr), the problem solving methodology (Mps) and the real world problem (P)) at the start of each research cycle needs some qualification.

At the start of the program all the participants knew that while expertise in the industrial use of open source might lie elsewhere in the commercial computing industry, it certainly wasn't visible within the OSS segment. The participants might also have expected the university researcher to already be an expert in open source and NGOSS but the truth was that the

researcher had not yet apprehended the framework of knowledge expressed elsewhere in this Thesis.

Similarly, the researchers early knowledge of the Action Research process could best be described in the terms of the Action Research Project Management Maturity Model ((*Estay-Niculcar, C. A. et al. 2002*) see Table 8, Section 2.1.7) as Level 1, Novice. Certainly an elaboration of the Action Research process in the terms of this Thesis could not have been attempted at the beginning of the OpenOSS Catalyst. However the later approaches to managing the risks in the program by developing multiple options for proceeding forwards (described in Section 2.2.4) would indicate at least a competence corresponding to Level 4, Managed.

Given the early lack of knowledge of the field, the researcher and sponsor were faced with a basic choice of whether to rapidly initiate a public program albeit with 'bounded rationality' or wait until the researcher knew a lot more. The decision to launch the program (while having limited knowledge) through a public TMForum 'birds of a feather' session took a degree of courage (or foolhardiness) on the part of the sponsors and the researcher. The whole approach could have been shot down in flames and made them a laughing stock in the TMForum. What actually happened was that the BOF attracted industrial participants who similarly knew a little but wanted to know more about open source. One might even say that the approach was accepted because 'in the land of the blind, the one eyed man is king'. Versions of an early paper (*Gallen, C. R. et al. 2005b*) were used to outline the perceived problem space and the proposed research process at a level which the researcher would now argue was a novice level of understanding. The paper provided an overview of problems in the OSS industry and the potential for open source but merely outlined Action Research in generic terms as described by (*Saunders, M., Lewis, P., & Thornhill, A. 2003*) and had limited reference to IS Action Research.

The forgoing discussion has important implications concerning the validity of the way in which the research results have been reported in this Thesis. (*Checkland, P. et al. 1998*) state that the research epistemology sets the terms in which what will count as knowledge from the research will be expressed. But what if the knowledge of the epistemology has matured in the interval between when the intervention occurred and the time when it is finally written up? (i.e. during the 'off-line' reflection of (*McArdle, K. L. et al. 2007*) discussed in Section 2.1.6).

In this Thesis, the view has been taken that the results of the program can be much better and more concisely explained if a common epistemology based on the most recent knowledge is used to set the context for all the results even from early in the program. Thus the results are presented as a dialectic between current theory and the findings of the set of projects. This is reflected in the organisation of the Thesis as explained in Section 1.10. For this reason the

background to interventions can sometimes reference papers published after the intervention occurred.

The approach is justified on two counts. Firstly on the basis that all of the participants in the projects had bounded knowledge of the industry and even of their own organisations. If papers or other sources have been subsequently discovered or published (particularly if published by a participant organisation as in the case of BT) which throw more light on the question, it is valid to use these as well. Secondly, subsequent sources can throw more light on the tacit knowledge shared during each intervention.

One of the benefits of Action Research is that it can give access to the tacit knowledge of the participants. The start of each project was characterised by many meetings to agree common goals in which the participants shared their experiences through presentations or position statements. In a number of cases participants alluded during the project (sometimes under a Non Disclosure Agreement (NDA)) to issues on which subsequently papers have been published or discovered which are referenced to increase the validity of the findings.

2.2.3 Entering the Problem Situation and Establishing Roles

During the course of the work, the researcher has had to deal with the access issues described in Sections 2.1.4 and 2.1.5 and gain access to three different situations.

The OpenOSS catalysts and NGOSSoil projects (Chapter 4) were conducted mostly under the umbrella of the TMForum. However meetings leading up to project initiation including early contacts with BT, Vodafone, Agilent and Nortel had to be conducted under separate NDA's including Invocom and the University. Once the program had been established the pre-existing legal framework provided by the TMForum greatly simplified this process.

The TMForum provides established practices concerning how members participate, contribute and address IPR (*TMForum 1989*). It also provides a standard form of contract called the Charter for Catalyst projects and defines a set of roles which the participants must staff including Team Leader, Service Provider Sponsor, Logistics Manager and Marketing Manager. The researcher shared a joint Team Leader role with Gary Bruce from BT who was also the Service Provider Sponsor. In practice this meant that the researcher concentrated on architectural and development issues while Gary Bruce concentrated on liaison with the other service providers. In the OpenOSS Catalyst the additional software development roles were also clearly defined in the Catalyst Charter (*TMForum 2004c*). The charter would be equivalent to a formal 'action document' (*Mumford, E. 2001*) discussed in Section 2.1.4.

While the TMForum provided a highly collaborative working environment a full understanding of its approach to open source does require an understanding of the power structures operating in the industry (*Avison, D. et al. 2007*) introduced in Section 2.1.4. This

issue is elaborated more fully in the description of the stakeholders in the OpenOSS project in Section 4.2.1 and in Appendix D.4.

A detailed discussion of the learning's from involvement with the OpenNMS project (*OpenNMS 2007c*) will be given in Chapter 6. Getting access to the OpenNMS development team was relatively easy in that the project is run with the objective of encouraging participants. No formal documentation needed to be put in place. It was however important to respect the fact that the three core leaders of the project (Tarus Balog, Dave Hustace and Matt Brozowski) also owned the OpenNMS group and made their living from the project. Significant efforts were put into building goodwill and a mutual understanding of what the research was trying to do. This included meeting the team face to face in North Carolina and at development conferences in order to keep them up to date with the work in the TMForum. In addition Arqiva agreed to pay for a week of consultancy from OpenNMS which provided some recompense for the effort the guys put in to help us.

Establishing access to Arqiva communications for the DVB-T TV project (Chapter 5) was more difficult. Arqiva were not yet members of the TMForum although as result of this research they recognised the relevance of the TMForum's standards to their work and their principal OSS suppliers are now members. In this case initial contacts were established more than a year before active engagement began. The researcher was able to leverage a remnant of trust derived from the fact that he had worked 10 years previously at Arqiva's former owner NTL. This at least opened to door for initial discussions. However several rounds of presentation to Arqiva's management were necessary before the work was allowed to proceed. These discussions of necessity included Invocom as the project sponsor and the University of Southampton. Because of the commercial sensitivity of the project, all parties signed NDA's.

When the proof of concept project was properly initiated, a statement of work was produced for the project and appended to the statement of work which Invocom proposed to commercially undertake for Arqiva. This would also be equivalent to the formal 'action document' (*Mumford, E. 2001*) discussed in Section 2.1.4.

In this case, Invocom were being employed as OSS consultants to assist Arqiva write an RFQ for the DVB-T OSS system. From an Action Research perspective, it was important to clearly establish roles in order to make clear that Invocom's work was not a deliverable from the researcher who was only focusing on the design of the proof of concept. It was also important to make clear that the researcher would eventually wish to publish results from the activities. This is an interesting example of managing the synergy and differences between consulting and Action Research discussed in Section 2.1.5.

2.2.4 Participative Action Research within the TMForum

The TMForum has represented a particularly challenging environment (client system) in which to work. Firstly, most TMForum activities (work streams or Catalyst projects) are resourced through fragile and short term alliances between members which are forged to move some mutually beneficial aspect of the work forwards. Agreeing common objectives and securing resources for a project is not a trivial activity and requires entrepreneurial and business development skills from the researcher. Secondly, members contribute on a 'best effort' basis and TMForum projects can be allowed to suffer if business priorities change. There is no guarantee that an initial project will follow through to a second phase.

Action Research is designed to be a fluid process which can respond to the changing complexities of the 'client system'. In order to minimise the risk to the overall program, throughout this research a portfolio of potential options for subsequent work activities has been developed which, while still addressing the overall research theme, accommodated the risk of any one project failing to materialise, falling apart (through disagreement) or being under resourced. This level of IS Action Research project management maturity corresponding to 'Level 4 Managed' (Table 8) introduced in Section 2.1.7, (*Estay-Niculcar, C. A. et al. 2002*).

In Appendix A there is a detailed chronology of the overall program in which the key business events (summary meetings, workshops, work activities or decision points) are described as interventions in the Action Research sense. To simplify explanation (and with hind sight) the chronology has been partitioned into the four identifiable overlapping projects summarised in Figure 4. However it should be noted that at the time when these activities were developed in response to market opportunities it was unclear which of these projects would proceed. Note also that the work with the OpenNMS project was common to all of these projects and is not split out separately.

In order to better illustrate the use of the Action Research process, the contribution of each intervention has been summarised in Appendix A according to the 5 phases of (*Susman, G. I. et al. 1978*) (Table 6) where phase 5, Specifying Learning, has often corresponded to a public presentation or a paper published about the work's findings. Note though, that it has not been possible to describe a clean partitioning of the work. In the real world, each intervention (meeting, workshop etc) may have covered more than one phase, or even more than one project and the research has followed a cyclical pattern of multiple iterations driven by external events rather than exhibiting a clean transition between phases. This is much as expected from the description of the mini cycles of Action Research by (*McKay, J. et al. 2007*).

2004				2005				2006				2007			
Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
	❶	❶	❶	❶	❶								❶		
				❷	❷	❷					❷	❷	❷		
					❸	❸	❸	❸	❸				❸		
		❹				❹	❹	❹	❹	❹	❹	❹	❹		

❶ Initial OpenOSS Catalyst Project

❷ Follow on activities from OpenOSS Catalyst Project

❸ Nortel contributed NGOSSoil project

❹ Arqiva DVB-T digital TV management project

Figure 4 Summary of chronology of projects during the Action Research program

2.2.5 Data

Several primary data sources have underpinned this research.

Firstly, many of the opinions and actions of the participants have been recorded in the agreed minutes from the TMForum meetings and workshops held throughout this project (or internal project minutes in the case of Arqiva). In many cases the researcher was also the minute taker but personal bias has been minimized by the fact that all the minutes and architecture documents were reviewed by the involved parties and in many cases the minutes were published for public scrutiny on the TMForum Collaborative work space. Often participants also shared position statements from their organisations in the form of presentation slides during these meetings.

Secondly, any material published in the public domain by the participants has also been used. Particularly in the case of BT this has taken the form of peer reviewed technical papers. In other instances this has drawn upon presentations given at TMW, press releases or other communications from the organisations in the public domain.

Thirdly, the technical descriptions in the Appendix were edited and/or authored by the researcher and reviewed for accuracy by the technical participants.

Finally throughout the project the researcher's personal log has been maintained with any additional relevant reflections on the state of the project or subsidiary technical investigations.

2.2.6 Ethics

The ethical conduct of the researcher represents an important consideration in an Action Research project (*Susman, G. I. et al. 1978*). Participants may be invited to share information which could be misinterpreted or used against them. It is important therefore that a relationship of trust is maintained throughout.

University based research offers many advantages in that organisations can be more willing to share information with a 'neutral party' such as a researcher at a university than they would with another company. However, although officially a student to the University of Southampton, the researcher was also sponsored by a small systems integrator, Invocom. To

protect the interests of Invocom and their customers, the University of Southampton, the researcher and Invocom were bound through back-to-back NDA's.

The researcher made a practice of explaining in all introductions that he did not just represent the University but also Invocom. In fact this was a useful relationship as Invocom's involvement opened a number of doors, and did not seem to present a hindrance at any point in the research. Early discussions with parties such as BT, Vodafone or Arqiva were covered by NDA's with Invocom which were 'back-to-back' with the University. In the case of Arqiva, the early work was particularly sensitive because Arqiva were in the process of bidding for a contract with the BBC which they subsequently won. Early discussions with Nortel were covered by a personal NDA between the researcher and Nortel.

The researcher also made a practice of explaining the nature of Action Research and that participants should expect that the process as well as the outcomes of projects would be the subject of study.

Working within the TMForum has brought several advantages. There are established practices concerning how members participate, contribute and address IPR (*TMForum 1989*). The TMForum recommends to members that they should not bring anything to meetings which they do not want more widely known. Fortunately, although early in the project material was shared with the researcher which was not in the public domain, the vast majority of the material referenced has since been put into the public domain by the participants through their participation in the TMForum and so is available to report in this research. In some specific instances special permission has been sought before publication.

A further ethical consideration was the use of students to support some of the work. Employing students during the summer was welcome but incorporating students' third year projects in the OpenOSS catalyst proved to be problematic. There were some concerns expressed by the University that too much pressure could be applied (perhaps indirectly) on students whose only obligation was to get their degree. It was recognized that participation in industrial research and presenting at industrial events clearly could have great educational benefit for students - but it could also have negative side effects on their other work.

A final ethical consideration for the researcher which is particularly problematic in an Action Research setting is to determine which work can justifiably be claimed as the researcher's unique contribution. In this case the researcher was a driving force behind the projects, but many others contributed to the outcomes. In addressing this issue it is perhaps helpful to consider again the distinction which (*McKay, J. et al. 2007*) make between Mps, the problem solving methodology (Open Source and collaborative development) which has been shared across the project teams and Mr (The research methodology and subsequent specified learning) which have remained the researchers sole responsibility and the core subject of this Thesis.

2.3 Summary

This chapter has laid out the reasons for choosing to use Participant Action Research in this project and has described how this research methodology has been applied. It has also described the key considerations which have determined the quality of the research design. The chronology of the over all program has been described and is further elaborated in 0. Finally some justification has been given for choosing to present the research in terms of a dialectic between the theory (epistemology) and the findings of a set of projects. Chapter 3 will complete the explanation of the research epistemology with a description of the OSS Industry and Open Source Problem Space.

Chapter 3 The OSS Industry and Open Source Problem Space (P)

Chapter 3 The OSS Industry and Open Source Problem Space

3.1 Chapter Context

Following the methodology laid out in Section 2.2.1, this Chapter 3, combined with Chapter 1, summarises, in terms which OSS industry players will recognise, a diagnosis of the Problems (P) which the industry is seeking to address through bodies such as the TMForum. It also provides a Framework of Ideas (F) which theorise how the collaborative innovation fostered in the standards organisations might be combined with open source and applied as a Problem Solving Methodology (Mps).

An initial understanding of the problem was published in an IEEE paper (*Gallen, C. R. et al. 2005b*), which was used early in the program to provoke discussion and educate the participants as to the rationale behind the interventions. The peer review process gave some confidence that the diagnosis was on the right track. The contents of this chapter have since been developed by means of literature review, market research, interviews, meetings and reflective learning. Consequently this chapter has been based on a snapshot of the industry situation using the best information available at the time of writing. As such it articulates a useful analysis and a set of questions which managers in the OSS industry will find helpful in considering whether to use, contribute to or initiate an open source project.

3.2 Introduction

(*Gacek, C., Lawrie, T., & Arief, B. 2001*) have noted that the open source phenomenon has attracted many interdisciplinary research contributions, drawing upon Computing Science, Management, Social Sciences, Psychology, Organisational theory, Economics and Law. It is therefore important to scope the study carefully in order to delimit the framework of ideas (F) which will inform the results of the research and to define a problem domain which can realistically be accessed in industry. The first Telemanagement World BOF presentation given to the TMForum[10] delimited the research around three questions which are elaborated in the following sections.

- Can open source contribute to the overall risk efficiency of telecoms OSS projects? (Section 3.3)
- Can open source contribute to the development of an OSS component market place? (Sections 3.4, 3.5)
- Can open source assist collaborative innovation between firms in the TMForum? (Section 3.6)

3.3 Risk Efficiency

(*Fitzgerald, B. & Kenny, T. 2003*) suggest that management resistance has limited the growth of open source within many organisations to the relatively invisible back office infrastructure where IT staff have been able to quietly deploy open source products because their low cost kept them below managements' budgetary radars. They suggest that it is very unlikely that

[10] 'Birds of a Feather' (BOF) session - 'Open source solutions for NGOSS' TMW Nice 21/5/2004

these products would have been sanctioned had management been made aware that vital infrastructure was to be based on software that was not supplied by a vendor with a legal responsibility for ensuring its performance and continued support.

The first proposals to initiate an open source program in the TMForum were met with some alarm and resistance from the COTS OSS vendors who made up the majority of the TMForum membership and it took considerable board level discussion to get approval to proceed[11]. It is generally perceived to be in the interests of COTS vendors to use FUD (Fear Uncertainty and Doubt) tactics (*Irwin, R. 1998*) to limit the penetration of open source (*Feller, J. & Fitzgerald, B. 2000*). Therefore a significant objective of this action research was to educate vendors and customers across the OSS industry as to the reality of the potential risks and benefits of open source.

3.3.1 Uncertainty Management

(*Ward, S. & Chapman, C. B. 2003*) propose that the term 'risk management' is inherently loaded with negative connotations concerned only with damage limitation. They argue instead for the term 'uncertainty management' as a corporate process which manages project uncertainty in order to plan to take advantage of potential opportunities as well as avoiding potential pitfalls.

The concept of 'risk efficiency' derives from financial portfolio theory where models originally proposed by (*Markowitz, H. M. 1959*) are applied to quantify the risk and opportunity posed by a portfolio of shares. (*Chapman, C. B. & Ward, S. 2004*) summarise how the concepts embodied in portfolio theory can usefully be applied to practical project management tasks without the unnecessary complexity of trying to use mean–variance quadratic programming. These guidelines are fleshed out in (*Chapman, C. B. & Ward, S. 2002*) and (*Chapman, C. B. & Ward, S. 2003*).

(*Chapman, C. B. et al. 2004*) define risk efficiency as

> 'the minimum risk decision choice for a given level of expected performance', 'expected performance' being a best estimate of what should happen on average, 'risk' being 'the possibility of adverse departures from expectations'.

The concepts of risk efficiency are illustrated in Figure 5. For any project, the overall Expected Cost is the best estimate of what the venture is projected to cost on average and the Cost Risk represents the possible unfavourable departures from expectations. If one assumes that there are a finite number of ways in which a project can be attempted which fall within the feasible solution area, each approach will carry a different Expected Cost and a different Cost Risk. If a project is badly planned then it will fall within the 'Area of Incompetence' (i.e. point A) but assuming good planning, the project is still faced with choices of approach which trade the Cost Risk against the Expected Cost.

[11] TMForum Board discussions reported to us by Martin Creaner, TMForum CTO.

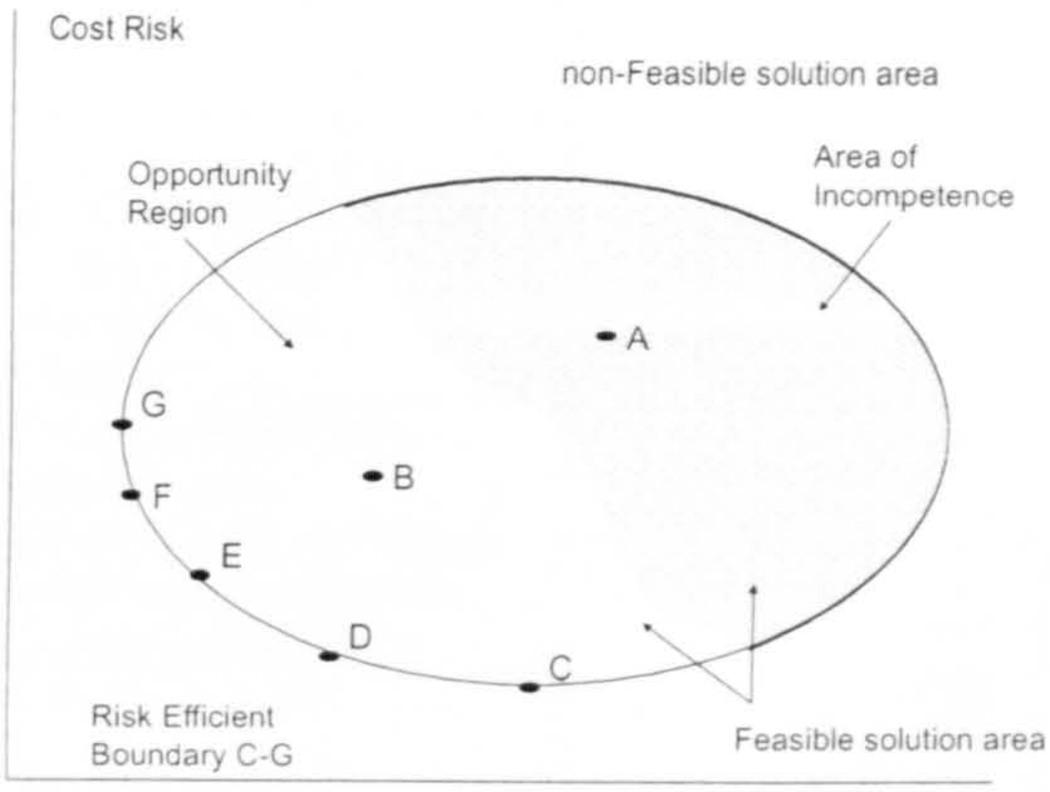


Figure 5 Risk Efficient options
(after Chapman, C. B. et al. 2004)

A project at point B is viable, but good management could still reduce its expected cost towards the Risk Efficient Boundary (C-G) thus the project at point B is in the 'Opportunity Region'. At the Risk Efficient Boundary there are no more actions which can make the project more risk efficient and it becomes a business choice to set the policy of whether to pay more to reduce the risk to a minimum or live with some risk and have the project projected to cost less. In practice, it is not possible to know where the risk efficient boundary really is but it is useful to understand that there is an Opportunity Region in which feasible projects could trade risk for expected cost.

Normal project management practices will seek to reduce the risk to a minimum within a given project's budget. If one considers a business as having a portfolio of projects, each with a different expected cost and a different business impact of failure, a company might be more profitable if it can choose to live with projects carrying a higher risk as long as it can live with the consequences of delay or failure; i.e. it can self insure and it will not go bust if the worst happens. This of course relies on the projects not being risk covariant; i.e. they are not all subject to the same risks and don't all go wrong at the same time for the same reasons.

The estimation of project cost should consider the entire life cycle of the project and not just the initial cost. If the project is directly tied to the introduction of a product or service, risk does not just apply to the cost of a project but also to the lifetime profitability of a project in a volatile market. This is important especially if the product is likely to follow a long adoption cycle as predicted by Moore (*Moore, G. A. 1999*), whose chasm theory describes a problem unique to high tech industries where new products initially sell well, mainly to a technically literate customer base, but then hit a lull as marketing professionals try to cross the chasm to mainstream buyers. Figure 6 shows how the chasm extends Rogers' (*Rogers, E. M. 2003*) well-known technology adoption curve profiling group adoption of new technologies.

Rogers' technology adoption curve (Rogers, E. M. 2003)	
group	profile
innovators	• Tend to seek out the new technology before it is formally marketed. • Innovators are typically very technology focused.
early adopters	• Accepts new products very early in the technology life cycle. • Early adopters find it easy to imagine, understand, and appreciate the new technology's potential benefits.
early majority	• Driven by a strong sense of practicality. • Want well-established references from relevant market segments before committing themselves.
late majority	• Shares all of the early majority's concerns and feel uncomfortable in their ability to manage the new technology. • Wait until a new technology has become a well-established standard.
laggards	• Want nothing to do with the new technology. • When they do accept a new technology, they typically do so without realizing it.

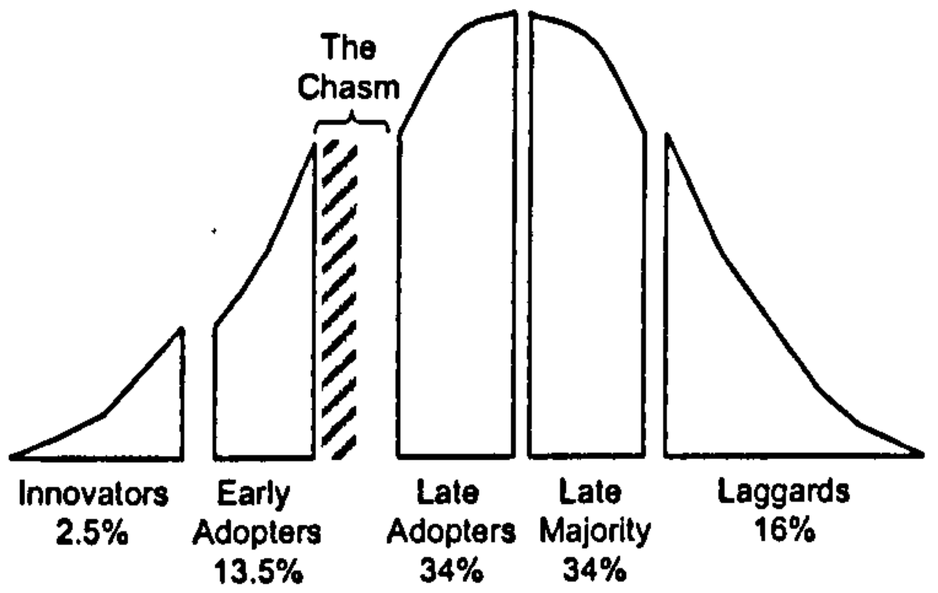


Figure 6 Technology Adoption Curve and Moore's chasm
(Rogers, E. M. 2003), (Moore, G. A. 1999)

The optimum risk efficiency of a telecommunications project is not constant but changes throughout the lifecycle of a project. The rate of market adoption of new telecoms services is very uncertain so one would expect the industry to wish to minimise the cost of incubating a portfolio of innovative services through to the early adopter stage, being less sensitive to service risk and more sensitive to expected cost at this stage. In other words, there is a need to find processes to maximise the number of innovative ideas which can be tested in the market at a minimal cost. Later if one of the products reached a tipping point (*Gladwell, M. 2000*), the industry would invest in reducing the risk commensurate with the importance of the revenue stream from the product.

The telecoms industry draws upon the many established processes for managing risk in large scale software projects (e.g. Prince 2 (*OGC 2002*)) and service providers often have large departments devoted to this task. Therefore in this discussion of project risks, it is not the intention to propose new or better software risk management practices. It is assumed that established best practices will be used to achieve the most risk efficient delivery of whichever development methodology is utilised. Rather, it is the intention to consider the effect which open source might have on the ability of the industry to choose the risk/expected-cost profile appropriate to the business case and the lifecycle of the project.

(*Chapman, C. B. et al. 2003*) offer a generic methodology for iteratively designing a risk efficient project through a comprehensive understanding of the sources of risk as delimited by a set of six questions (Who, Why, What, Which way, Wherewithal, When). The Zachman

Framework (*Zachman, J. A. 1987*), (*Sowa, J. & Zachman, J. A. 1992*) delimits systems analysis in terms of a very similar set of questions (Who, Why, What, How, When and Where). The NGOSS lifecycle methodology (*TMForum 2004a*) is based upon Zachman and a framework for analysing open source development based upon Zachman has also been proposed by (*Feller, J. et al. 2000*).

Thus it seems reasonable to adapt Chapman's methodology to construct the risk analysis framework outlined in Table 10 which considers the OSS Component Market place, the OSS Development Approach and the Impact on Practice for each of the stakeholders in the supply chain, as open source is introduced to an OSS COTS environment. This analysis framework is used to help structure the analysis of the opportunities for open source in OSS through the rest of the Thesis. A suggested overall Impact on Practice for each type of industry stakeholder is summarised in Chapter 7.

Using Chapman's methodology provides the advantage that this analysis fits within a wider body of consistent project management literature which could help industry players operationalise the analysis in real projects. For instance it could help in providing a basis for equitable contract design between players in the open source supply chain. (*Chapman, C. B. et al. 2003 p 142 ff*).

Table 10 Analysis Framework

Broad Concerns	**Project Analysis** (Using 6 W's - Chapman, C. B. & Ward, S. 2003 page 4)			
OSS Component Market Place Examination of the motivations and roles of the OSS market players and how this might be influenced by the introduction of open source	**Who (parties)** Who are the parties ultimately involved?	SP's	Integrators	ISV's
	Why (motives) What do the parties want to achieve?	?	?	?
Development Approach Consider differences in development approaches; (Bespoke Development, COTS, and Open Source)	**What (design)** What is it that the parties are interested in?	?	?	?
Impact on Practice Special measures required to successfully deliver a system using the chosen Development Approach What changes would be required in the way projects were approached to incorporate an open source methodology?	**Which way (activities)** How is it to be done?	?	?	?
	Wherewithal (resources) What resources are required?	?	?	?
	When (timetable) When does it have to be done?	?	?	?

3.3.2 Better Investment Decisions

Perhaps one of the reasons for the emergence of open source contributors and champions from within corporations is that it allows engineers to bypass the frustrations they feel at investment rules which discount their professional judgement. Benkler has described open source as an example of 'commons-based peer production' which is an alternative means to organise economic production emerging in the networked economy. This offers a mechanism for more efficiently allocating resources to projects than the traditional business case based on a firm's reading of market signals (*Benkler, Y. 2006*), (*Benkler, Y. 2002*). In this model users and

developers self allocate themselves to the projects in which they have most confidence and where they feel they have the most to gain or contribute.

In Section 3.6.1 we will discuss how many firms experience an uneasy relationship between their R&D and Marketing functions. This disagreement may in part arise from the consideration that the technology investment rules operating in most organisations make it difficult to justify strategic technology investments. It is important therefore to consider how the introduction of open source could be reflected in a firm's strategic budgetary processes.

A survey of the capital budgeting practices at large American corporations (*Farragher, E. J., Kleiman, R. T., & Sahu, A. P. 1999b*) found that 78% of the respondents used Net Present Value (NPV) as a primary criterion when evaluating capital investments and that 45% percent of the firms reported a policy of rejecting a capital investment opportunity that has a favourable 'strategic analysis' if the NPV of the opportunity was negative. Under these rules, a business unit must justify an investment decision by projecting a return above a 'hurdle rate' (cost of capital) which is the minimum rate of return that must be met for the corporation to allow a business unit to undertake a particular project.

A risk premium may also be attached to the hurdle rate if management feels that specific opportunities inherently contain more risk than others that could be pursued with the same resources. (*Farragher, E. J., Kleiman, R. T., & Sahu, A. P. 1999a*) noted that raising the required rate of return was the most popular method for making a formal quantitative risk adjustment. (*Dixit, A. K. & Pindyck, R. S. 1994 p7*) note that many companies set the hurdle rate 3-4 times higher than would be expected by the prevailing interest rates - which they explain as an effect of the firm heuristically taking into account the opportunity cost of not being able to invest in other projects. However a problem with using a hurdle rate that is too large is that profitable projects may be turned down and it will bias the firm in favour of short-term returns over long-term investment with an uncertain return.

(*Trigeorgis, L. 1996*) makes the case that NPV systematically understates the value of an investment made in an uncertain environment. If the market opportunity for the technology is uncertain, a normal NPV calculation would deter the firm from making the investment.

An alternative approach to evaluating investment decisions is the use of Real Options which draws upon share portfolio option pricing theory. A real option is the right, but not the obligation, to undertake some business decision in the future. A firm may 'purchase' a real option to sell a technology in the future by investing in R&D in the present.

Unfortunately calculating the value of a real option is quite complex and although standard texts such as (*Dixit, A. K. et al. 1994*) and (*Trigeorgis, L. 1996*) provide guidance as to how to calculate the value of Real Options, (*Latimore, D. 2002*) of IBM notes that Real Options are not yet universally recognized as a means of valuing capital investments but considers that determining the exact value of a real option is not necessarily critical. Instead, understanding the drivers of the valuation and the value relative to traditional methods is much more

important. The most important observation being that as uncertainty increases, so does the value of a real option.

(*Thorogood, A. & Yetton, P. 2004*) have applied a real option approach to complex IT infrastructure decisions and demonstrated its application in an action research project which looked at the introduction of a complex CRM system into a bank (*Reynolds, P., Thorogood, A., & Yetton, P. 2005*).

(*Thorogood, A. et al. 2004*) suggest that most IT infrastructure investment decisions are made on the basis that the infrastructure will enable many applications; each of which will have its own return on investment. The technical case for the investment may be self evident to the technologists as it allows them to have a consistent architecture and avoid piecemeal infrastructure growth. However the business case for the infrastructure investment still has to be assembled by summing the NPV contributions of the envisaged individual applications until the case is made (or made up?!).

The unfortunate practical result of this approach is that the resulting project is often overly complex and poorly planned because its design must include from day one all the applications and benefits which have been promised. Thus poor and rapid requirements capture, or the promise of 'too much too soon', leads to bad implementation choices. Furthermore, during the life of the project it may turn out that other applications would be better candidates for development but these are beyond the scope of the project (or subject to variation orders) and so can not be considered.

(*Thorogood, A. et al. 2004*) argue that the justification of a major infrastructure investment should be treated separately from that of the individual applications which leverage it by using an option calculation which recognises the value which the firm derives from the flexibility to deliver business applications as and when they are needed. The individual projects which leverage the infrastructure should then be separately planned based on the specific NPV of the investment at that time. By using these mechanisms they argue that firms can avoid the clash between the 'business-only' and 'technical-only' framing of the infrastructure problem.

(*Thorogood, A. et al. 2004*) also note that companies often resort to allocating budgets to the CIO for discretionary expenditure because it has consistently been a problem to justify the cost of experimentation to determine the value of future technology. They suggest that this process could be thought through in a more structured manner and justified within a Real Options framework due to the many options created including the options to scale up or down, outsource or abandon a project.

Although (*Thorogood, A. et al. 2004*) argue from a banking infrastructure perspective (and banks tend to understand financial options better than most!), their observations have several bearings on the present problem. The introduction of an NGOSS architecture to a service provider is essentially an infrastructure problem where the benefits derived by the individual applications of NGOSS can only be realised if an NGOSS infrastructure is first put in place.

Therefore, potential adopters of NGOSS could perhaps benefit from consideration of the investment as creating a portfolio of options in flexible service delivery rather than trying to justify it on a specific cumulative application basis. Large ISV's and equipment vendors who find it hard to justify the introduction of a new platform infrastructure technology across the business (based on the cumulative benefit which each product may derive from the infrastructure), may also find that an options based business case is more useful.

(*Thorogood, A. et al. 2004*) note that technically their strategy can be realised by using standards such as web services to decouple the individual applications and rapid application development and prototyping to reduce the uncertainty of each application. NGOSS could support a telecoms implementation of their approach to strategic investment. The introduction of open source takes this argument further. Firstly, open source could facilitate prototyping. Because of its zero up-front cost, the cash flow associated with using an existing open source project makes it very attractive for the early stages of infrastructure development.

Secondly, rather then invest in a feature rich COTS solution before the features are needed by a specific application, it may be more cost effective to meet present needs within an active open source project (which has the potential to develop the features), and provide encouragement for the community to move in that direction. This is a particularly attractive option if the project is implementing a standard which would allow it to be replaced by a COTS product later if absolutely necessary.

Thirdly, the arguments made by (*Thorogood, A. et al. 2004*) for an options approach to allocating a CIO's discretional expenditure may deliver more value when applied to an open source project. (*Baldwin, C. Y. & Clark, K. B. 2006*) have demonstrated that well architected open source code has an option value in that it offers more opportunities for an adopter to adapt it to novel applications. An open source project which has many potential applications has a high option value for its adopters and creators. An organisational strategy favourable towards open source represents an option to profit from the portfolio of projects which the open source community represents.

Finally, macro business decisions can result in the discontinuation of projects which often results in an inventory of unused code which is very demoralising for its creators and wasteful of their work. However the unexpected community uptake of dormant projects has been noted by (*Fink, M. 2002 p169 ff.*). By making code open source it becomes possible for a wider community to keep it alive if the original level of investment is not continued. Because an open source contribution can be resourced at varying levels, open source provides an option for the development of a feature to be accelerated, slowed or postponed to match the needs of the market.

3.3.3 Summarising Open Source Uncertainty Management

This section has considered some concepts (summarised in Table 11) which players in the OSS industry might use to analyse and manage the uncertainty of open source. The action research interventions have been designed using the model outlined in Table 10 to examine the OSS Component Market place and the OSS Development Approach with a view to reaching some conclusions and recommendations as to the Impact on Practice for the industry to adopt. The next section will consider the first analysis dimension in Table 10, the OSS Component Marketplace.

Table 11 Summarising Risk Efficiency Concepts
1. Chapman's risk efficiency approach to project management might offer a simple framework for analysing the behaviour of the players in real open source projects and managing the risks they represent.
2. Risk efficiency concepts may also provide a vehicle for IT managers to better quantify the investment risks and rewards associated with using open source alongside COTS systems in large projects.
3. The decision to invest in an NGOSS infrastructure may never be made if it requires justification based simply on an NPV analysis. A Real Options approach may put a more realistic value on the flexibility which an NGOSS infrastructure provides.
4. Open source can provide a means to operationalise a Real Options approach by making it possible for adopters to extend open source products in a way which is closed to them under the COTS model.

3.4 OSS Component Market Place

(*Hopkins, J. 2000*) defines a software component as 'a physical packaging of executable software with a well defined and published interface'. While this provides a simple technical definition, it does not consider the business aspects of a software component as a Commercial Off-The-Shelf (COTS) product. (*Morisio, M. & Torchiano, M. 2002*) list several alternative definitions of COTS products but for our purposes the SEI COTS-Based System Initiative provides a usable definition which defines a COTS product as;

> "A product that is sold, leased, or licensed to the general public, offered by a vendor trying to profit from it, supported and evolved by the vendor, who retains the intellectual property rights, available in multiple identical copies and used without source code modification." (*Brownsword, L., Oberndorf, T., & Sledge, C. A. 2000*).

The majority of the stakeholders in the current OSS market place deal in COTS components which would fit this definition.

The OSS market can be thought of in several dimensions both in terms of the players in the supply chain and in terms of the types of software components produced. (*Vitharana, P. 2003*) defines three stakeholder types in the component based software engineering supply chain, Component Developers, Application Assemblers and Customers, noting that firms may function in more than one capacity. The following sections will consider some more general characteristics of the telecommunications industry before looking specifically at each of these stakeholder types.

3.4.1 Network Effects in the Telecoms Industry

Although it is beyond the scope of this Thesis to attempt a full economic analysis of the OSS industry, the insights of several economists are helpful in understanding the inter-relationship between players and the role of standards in the industry.

The telecommunications industry exhibits what economists call strong network effects at every level (*Economides, N. 1996*). A network industry (in the economic sense) consists of products or services which can be characterised as offering *complementary* or *substitute* functions. A telecommunications service (say a telephone call) demanded by a consumer will be composed of many complementary functions, often from different service providers, all of which must be present for the service to be available. Complementarity can be achieved only through the adherence to specific technical *compatibility* standards. However compatibility also makes it possible for a competing service provider to offer a substitute service of equal utility to the consumer.

A second characteristic of a network industry is the fact that the value of a unit of a good increases with the number of units sold [12], an observation called a 'positive network externality'. In telecommunications, the more consumers that can access a network service, the more valuable it becomes. This is a significant driver for open interconnectivity standards between service providers and between equipment vendors in the transport and service control layers. Increasingly open interconnectivity is also becoming important in the OSS layers as service providers need to electronically broker and monitor service level agreements on behalf of higher value clients.

Firms in a network industry need to make strategic choices about which technical compatibility standards they will support based upon their estimate of the equilibrium market structure which will result (*Economides, N. & Skrzypacz, A. 2003*). In making this choice it is important to contrast the benefits associated with the exploitation of network externalities versus the exploitation of monopoly pricing power (*Economides, N. & Flyer, P. 1998*). On the one hand, a firm can choose to make its output compatible with an established standard, as this increases the value of the product to the customer. On the other hand, by making its output incompatible with other products, the firm can gain monopoly power, even though its output is less valuable to the customer.

In our context the concepts of a network industry apply equally to the telecoms services and to the telecoms equipment and OSS applications which enable them. The concepts can also be applied to an economic analysis of open source.

(*Mustonen, M. 2005*) has noted that firms may contribute to open source programs if the open source program is a complement to the firm's products or if supporting an open source program hurts the firm's competitors. Such costly actions allow the firm to influence the direction of the programming towards compatibility and also to learn the direction of the project. However Mustonen also found examples of firms creating competition by supporting open source communities, whose programs are substitutes to the firms' programs (i.e. open

[12] Or more accurately, 'the value of a unit of the good increases with the expected number of units to be sold.' The demand slopes downward but shifts upward with increases in the number of units expected to be sold (*Economides, N. 1996*)

source versions of their own products). Mustonen's economic model attempted to explain this by showing that when support creates compatibility between the programs and they exhibit network effects, a software firm may support a zero-priced open source program even if it is a substitute. The firm benefits from adapting and growing the market opportunity for its own product.

The most visible public debates about open source have tended to concern the rivalry between Microsoft Windows and Linux - both systems having great economic importance with very large developer communities and installed bases. (*Casadesus-Masanell, R. & Ghemawat, P. 2006*) have modelled the markets for Linux and Windows and have suggested that both systems can continue to co-exist if Microsoft resist the temptation to treat Windows as a cash cow and find ways to encourage external developers to use, modify and contribute to their technology. In an interview with (*Silverthorne, S. 2005*), they suggest this might be done by making more code available to view and being more responsive to feature requests and bug fixes. However their key contribution to the debate is in considering the effects of '*demand side learning*' by which sophisticated user/programmers can easily learn how to use the software and modify it to their own purposes. Open source development is thought to harness demand-side learning more effectively than traditional 'closed' models.

Microsoft have argued that most customers don't want access to their code (*Matusow, J. 2005*) but (*Kuan, J. 2002*) has suggested that in certain markets the consumer choice between open source and proprietary software is a form of make or buy decision. Where most high-paying users are also programmers, open source or community organisation would be more likely (e.g. engineering tools or utilities), whereas if most high-paying users were non-programmers, proprietary or closed organisation would be more probable (e.g. word processors and spreadsheets).

Since OSS components have a strong dependency on complementary components, one would expect ISV's to be motivated to make it easy for end users to integrate their components with other vendors' products. One would also expect ISV's to make it easy for the developers of complementary components to understand how to integrate with their products. Furthermore, in this market, the end users are highly sophisticated and potentially open to taking advantage of 'demand side learning' facilitated by open source. Thus there is a case for considering whether the adoption of open source could benefit all of the actors in the market.

Even if the whole market didn't move to open source, the work of (*Casadesus-Masanell, R. et al. 2006*) suggests that, as in the case of Microsoft, the entry of some open source components might also change the behaviour of the other closed source vendors - encouraging them to make more information about their products available in order to reduce the cost of integration.

3.4.2 Customers - Telecoms Service Providers

Having looked generally at the network effects in the industry, this section will consider more specifically the current state and behaviour of the customers in the OSS supply chain, the telecommunications service providers.

3.4.2.1 Outsourcing of Functions

The business functions and processes within telecommunications service providers are characterised by the TMForum using the eTOM (*TMForum 2007b*). In his treaties on the nature of the firm, (*Coase, R. A. 1937*) argued that business functions fundamentally exist within firms and not on the open market because the transaction costs are too high to use an open market to organise resources except at levels of large granularity. Porter's value chain (*Porter, M. E. 1998*) popularised the notion of maximising efficiency by fully understanding how each function in a firm adds value to the firms output. The emergence of e-commerce is making it increasingly possible for service providers to cost effectively componentise individual business functions and offer them as services to other service providers or alternatively outsource them completely.

This is leading to the entrance of players from other markets (such as Tesco or Virgin in the UK) as Virtual Network Operators (VNO's) who own the customer relationship and use the infrastructure of service providers. Regulatory pressure is also forcing the dis-aggregation of network services through, for instance, unbundled access to the DSL local loop. This in turn is applying pressure to favour more generic B2B integration technologies (such as web services) over telecoms specific solutions.

As an example of the issues raised when applying B2B technology to OSS, BT have described the difficulties they faced in integrating a wholesale DSL service using ebXML. (*Duke, A., Davies, J., & Richardson, M. 2005*), (*Duke, A., Richardson, M., Watkins, S., Wahler, A., & Schreder, B. 2006*). Although these papers are making a case for using ontology technology to help with the B2B integration, many of the issues described might have been more simply addressed through an open source interface implementation shared between the parties. COVAD, one of the OpenOSS catalyst sponsors created an open source OSS/J trouble ticket implementation for integrating with ILEC's across the USA (*COVAD 2007*).

The trend for service providers to concentrate on the core business has also lead to outsourcing much of the R&D previously done internally and also to the outsourcing of integration functions on a per project basis to SI's like Invocom. Fransman has contributed a comprehensive study of the telecoms industry through the 1999 recession (*Fransman, M. 2002b*), (*Fransman, M. 2001*), (*Fransman, M. 2002a*). Fransman notes an R&D transformation in the 1990's whereby in the 'old telecoms industry' the 'innovative engine' was located largely

in the central research laboratories of the monopoly network operators[13]. By 1999 in the 'new telecoms industry', the 'R&D engine' had moved decisively into the specialist technology suppliers. As an example, Fransman notes that, in 1999 five top incumbents - AT&T, BT, Deutsche Telekom, France Telecom and NTT on average allocated only 2.6% of their sales to R&D suggesting that these telecoms operators were only slightly more R&D-intensive than the beverages industry.

However Fransman warns against confusing R&D with innovation suggesting that firms with low R&D intensity may nevertheless be highly innovative, and their innovativeness may lead to competitiveness and high profitability. Since all of the network operators are very dependent on specialist technology suppliers, much of their learning takes the form of *learning to use, rather than learning to produce*, technology - surely an example of 'demand side learning' (*Casadesus-Masanell, R. et al. 2006*) discussed in Section 3.4.1.

3.4.2.2 Location of R&D Resources

Looking more specifically at OSS development, a study of the OSS market (*Goldman, L. 2004b*) has revealed that the global spending on Operational Support Systems within the telecoms industry is around $30 billion dollars per year (about 3% of total telecoms revenue) with about two thirds of this being spent on custom developed systems and one third on COTS software. Some 200,000 IT professionals are employed globally by service provider IT departments, many of whom are working on internally developed solutions. The majority of the $10 billion dollars COTS OSS market is accounted for by only the top 100 service providers and when deployed by these service providers the COTS products are often heavily modified before use. Against this background, it can be seen that the lions' share of OSS development activity in the telecoms industry is still done internally by service providers. Over the last 20 years, COTS solutions have made significant inroads but these solutions still only account for one third of global OSS software development.

Continued internal development within service providers was confirmed in this study by several visits. For instance a visit to a German Tier 1 service provider T-Systems[14] revealed that they were using open source components internally to help create sophisticated MPLS provisioning systems which they could not find in the market place. However, their management were raising questions about the long term sustainability of an approach using only internal development. Other operators such as T-Mobile in the UK [15] had used external consultants to review an internally developed inventory system and had concluded that this was strategically better than COTS alternatives.

---

[13] Examples of major Telco labs were AT&T's Bell Labs, BT's Martlesham laboratories, France Telecom's CNET, NTT's electrical communications laboratories etc.

[14] Private Open OSS discussion with T-Systems in Germany 1 October 2004

[15] Private Open OSS discussion with T-Mobile in UK 13 June 2006

OSS Observer (*Goldman, L. 2004a*) further reports that in developing countries such as India, Indonesia and Vietnam, low labour rates limit the commercial OSS opportunities for global OSS ISV's. In many of these countries, alternative low cost software solutions have emerged because operators are unwilling or unable to afford the price of the applications used by operators in the West. These solutions are often supplied internally or from other emerging countries; a fact evidenced by several TM Forum Catalyst projects which have showcased non-mainstream capabilities originating in the CIS or India. Global market surveys do not necessarily reflect the contribution these solutions make because they are internal or low budget (in Dollars).

3.4.2.3 Telecommunications Industry Layer Model

In the developed world users of OSS systems can be characterised in three tiers. In the primary tier are the large national and translational incumbents many of which are privatised PTTs. In the secondary tier are smaller alternative operators who mostly compete with the primary tier in the high value business market. In the third tier are niche players who target very specific opportunities – often adding value on top of the wholesale infrastructure of the incumbents. A simple characterisation of these tiers is given in Table 12.

Table 12 OSS purchasing patterns of Telecoms operators

Market	*Internal R&D*	*Product Purchasing*	*Other factors*
1st Tier (BT, AT&T)	• Some Internal R&D • Historically used internal solutions • Big integration & support teams	• May use set of 'best in Class' COTS products • Will modify selected products extensively • Often do own integration	• Have a historical legacy of 100's of systems to integrate with new services
2nd Tier Alternative Operators *(C&W, COLT)*	• Limited or no R&D.	• Will use set of 'best in Class' COTS products • Will spend on products but not on R&D • Often partner with large SI's for integration	• Finding they have created a legacy architecture through hasty start-up purchasing decisions
3rd Tier Niche (Truphone)	• No R&D but will use any pragmatic solution that works.	• Tend to use enterprise solutions • Open to pragmatic open source or free solutions • Very flexible approach • Limited capital spend	• Niche markets can be highly profitable with flexible bespoke solutions not easily offered by large SP's
Emerging Markets	• May internally develop since labour is cheap vs. COTS	• Often purchase locally or in a lo-cost market because they cannot afford Tier 1 solutions.	• Have clever staff able to create local workable solutions.

In general the market for COTS OSS systems is a relatively small pool compared to other markets for commercial software such as e-business. This is evidenced by the fact that the Telemanagement Forum, which is the premier industry body supporting Service Provider OSS standardisation, has only 600 corporate members including suppliers.

Fransman has proposed a telecommunications layer model for categorising the vertically integrated players in the telecoms industry (*Fransman, M. 2001*), (*Fransman, M. 2002a*). In layering the industry, Fransman considers network services presented through the TCP/IP interface as the key enabler for non-traditional telecoms players to enter the industry - although

telecoms players often offer services in these domains as well. While far from a comprehensive market analysis, it is informative to consider the general penetration of open source across these layers as illustrated in Table 13. This should be considered both in terms of the availability and variety of open source projects and the market share.

Table 13 Telecoms industry layer model and open source penetration

Telecoms Industry Layer Model (Fransman, M. 2001)			**Open Source Applications**		
Layer	**Activity**	**Example Companies**	**Variety**	**Market Share**	**Examples**
VI	***Customers / Consuming***		Medium	Low	Multiple end user applications • Mozilla Firefox • Open Office • Helix (Real player)
V	***Applications Layer, including contents packaging*** (e.g. Web design, on-line information services, broadcasting services, e-commerce etc)	Bloombergs, Reuters, AOLTime Warner, MSN, Newscorp, etc	Medium	Medium	Content management /authoring applications • Plone/Zope (used by Viacom) • ComPeire / Sugar CRM • Icecast - streaming multimedia • Blender - CGI animation Tools for building sites • LAMP (Linux, Apache, Php, Mysql) • JBoss JEMS • Exo Portal
IV	***Navigation & Middleware Layer*** (e.g. browsers, portals, search engines, directory assistance, security, electronic payment, etc)	Google, Yahoo, Netscape, Vizzavi, Genie, etc	Medium	Medium	Open source infrastructure • Apache / Tomcat • Support for bespoke development
III	***Connectivity Layer*** (e.g. Internet access, Web hosting)	IAPs and ISPs ZenInternet RackSpace	High	High	Open source infrastructure • Apache / Tomcat • DNS / BIND • Linux • Network Management Tools
TCP/IP INTERFACE					
II	***Network Layer*** (e.g. optical fibre network, mobile network, DSL local network, radio access network, Ethernet, frame relay, ISDN, ATM, etc)	AT&T, BT, NTT, WorldCom, Qwest, Colt, Energis, Vodafone, NTT DoCoMo etc	low	Research	Next generation networks • Focus SER - Sip Express Router • Focus OpenIMS • Mobicents Middleware • ObjectWeb Consortium Other • Network Management Tools
I	***Equipment & Software Layer*** (e.g. switches, transmission equipment, base stations, routers, servers, CPE, billing software etc)	Nortel, Lucent, Cisco, Ericsson, Nokia, etc	Unknown	Embedded	Open source used as embedded components

Open Source applications appear predominantly above the TCP/IP demarcation. At the customer layer, Microsoft has a dominant market penetration on the desktop[16] but is loosing ground in the corporate space to Open Office[17]. In addition fully featured open source SME business systems are becoming viable (*Locke, J. 2004*).

Open source is known to be used extensively by Google in the Navigation & Middleware Layer (*Moody, G. 2006*). In fact Google is actively encouraging development using the open source model through sponsoring many projects including the sophisticated Google Web Toolkit (GWT) (*Hanson, R. & Tacy, A. 2007*), and sponsoring students contributions to Google's 'summer of code'[18]. All engineers in the company are actively encouraged to take one day a week to work on their own pet projects which can be open source (*Elgin, B. 2005*).

The majority of sites in the Applications layer are driven by Apache[19] with the LAMP stack being used to develop many of these sites (*Lee, J. & Ware, B. 2003*). Although having lower market penetration, fully featured open source applications are available in this layer including Application servers (JBoss, Jonas, Zope), Portals (JBoss Portal, Exo Portal), Content Management (Plone) and Customer Relationship Management (SugarCRM, ComPiere).

Furthermore, players in other specialist industries have reported very significant savings in large scale open source integration projects. Fitzgerald reports on an open source integration project in Beaumont hospital in Dublin with a projected 5 year cost of approximately €600,000 euros representing savings of greater than €6M euros compared to the COTS equivalent solution (*Fitzgerald, B. & Kenny, T. 2004*) (*Fitzgerald, B. et al. 2003*). This suggests that ignoring the open source option could be a costly missed opportunity for service providers.

Open source has most penetration in the Connectivity layer where generally Linux or BSD based IP stacks power internet connectivity. Service providers with business interests above the TCP/IP demarcation also have interests in open source. For instance the C&W Internet Solution Centre in Munich was a long term sponsor of the OpenPKG project until 2005 (*OpenPKG*).

On the other hand below the TCP/IP interface demarcation, Open Source penetration is much less visible in the service providers, being mainly confined to research projects or collaborative implementations of next generation network technology. Table 14 lists some of the key projects which have telecommunications sponsorship. However these are mostly projects which deliver services above the IP layer. The fact that players appear not to be actively using open source below the TCP/IP demarcation is an interesting observation, suggesting that open

---

[16] Web Browsers: April 2005 Market Share : Internet Explorer 83.7% Firefox 10.28% (doubled since 2004) Source: Enid Burns | April 26, 2005, 'Firefox Market Share Gains Continue' http://www.clickz.com/showPage.html?page=3500691

[17] Microsoft Office retains 95% of the general market, OpenOffice.org and StarOffice have secured 14% of the large enterprise market as of 2004 and 19% of the small to midsize business market in 2005. The OpenOffice.org web site reports more than 62.5 million downloads. http://en.wikipedia.org/wiki/OpenOffice.org#_note-29 [Accessed June 2007]

[18] http://code.google.com/oss.html [Accessed 23-8-07]

[19] http://news.netcraft.com/archives/web_server_survey.html June 2007 Web Server Survey: Apache 53.76%, Microsoft 31.83% [Accessed June 2007]

source may be very under-utilised as a development technique in the service providers when compared with significant new entrants such as Google.

However a key observation is that while service providers have not created an open source eco-system for OSS solutions, they are willing to use a selection of open source resource management tools both for network administration (*Kretchmar, J. M. 2004*) and for security management (*Howlett, T. 2005*) (*Lawton, G. 2002*) (*Koziol, J. 2003*). The key point being that these are often stand alone tools for which network administrators do not have to get budgetary approval. They are generally not integrated into any higher level OSS architecture.

Table 14 Prominent telecoms specific open source projects	
Focus SER Sip Express Router	SIP Express Router (SER) is a high-performance, configurable, free VoIP SIP server (*IPTEL 2007b*)
Focus SEMS SIP Express Media Server	SEMS is a SIP media server featuring announcement, voicemail, conference, ISDN gateway, IVR and more. SEMS works together with SER, the Sip Express Router (*IPTEL 2007a*)
Focus OpenIMS	Open Source IMS Core consists of Call Session Control Functions (CSCFs), the central routing elements for any IMS signalling, and a Home Subscriber Server (HSS) to manage user profiles and associated routing rules (*FOKUS 2007*), (*Magedanz, T., Witaszek, D., & Knuettel, K. 2005*)
Mobicents	Mobicents is an Open Source JAIN SLEE 1.0 Certified product. Mobicents is a Java VoIP middleware platform, which is generically applicable to a variety of problems demanding high volume, low latency signalling. Examples include call control, online gaming, financial trading, sensor network integration and distributed control. (*Mobicents 2007*)
ObjectWeb Consortium	In January 2002, ObjectWeb started as a joint project between INRIA, Bull and France Telecom in order to create open source middleware. ObjectWeb sponsors multiple projects like the eXo portal server and JOnAS J2EE app server (*OW2 2007a*)

3.4.3 Component Developers

(*Morisio, M. et al. 2002*) propose a number of dimensions in which COTS components can be characterised. One of these dimensions is the Architectural Level at which the system contributes as illustrated in Table 15 which also lists example COTS and open source components which could be considered as possible fits into each of the OSS component categories.

Most COTS OSS systems currently in the market place would be positioned at the 'Core' Architectural Level i.e. 'products which provide domain specific functionalities' in the OSS domain. It is important to realise that these core products are usually architected using a unique combination of components drawn from the other categories.

OSS component developers fall broadly into four categories described below.

Table 15 Architectural level of COTS
(after Morisio, M. & Torchiano, M. 2002)

Architectural Level	OSS Examples	COTS OSS	Open Source candidates
Core Products which provide domain specific functionalities;	CRM / Trouble Ticket	Clarify Remedy	Sugar CRM RT OTRS ComPiere
	Service Assurance	HP OpenView IBM Netcool, Agilent NgnAS	OpenNMS
	Inventory	Cramer	OpenPegasus (DMTF CIM)
	Service Activation	Metasolve ASAP	
	Resource (Element) Management	Nortel, Cisco, etc	Various open source resource management tools
Support Elements that cover a well defined and standardised role in the architecture but do not provide vertical functionality;	Rules Engine	ILOG Rules	JBoss Drools
	Workflow engine	BizTalk server	JBoss Jbpm
	Database	Oracle Versant (Object database)	PostgreSQL MySQL Hibernate
	Other	Nokia OpenEMS	OpenNMS AndroMDA ModelWizard
User Interface Highly reusable user interface components.		ILOG Telecoms Graphics Objects[20]	Java Server Faces Swing Netbeans Platform
Middleware Software which enables communication and integration;	Application Server platforms	Microsoft .NET IBM Websphere BEA	JBoss J2EE Jonas J2EE Sun GlassFish J2EE
Operating System		Microsoft NT Various UNIX	Linux Open Solaris

3.4.3.1 Telecoms Equipment Vendors

Telecoms Equipment Vendors (e.g. Lucent, Nortel, Alcatel, and Cisco) provide domain specific capabilities which aid the deployment and use of their elements. Usually this is in the form of element managers or other capabilities in the resource management layer. Equipment vendors are motivated to develop OSS capabilities to increase the volume of equipment sales by making the customers life easier. OSS development is seen as a cost.

The Telecoms Equipment vendors are generally organised to deliver truly productised element management solutions which go through the same waterfall development process and testing regime as the elements they manage. For this reason equipment vendors are generally less flexible than OSS ISV's and very loathe to customise an element manager or provide customers with 'specials'. However large network sales are often predicated on the development of interfaces to integrate into the customer's OSS. This is seen as a cost which they would rather not incur.

It is interesting to note that prior to the dot.com bubble bursting, one strategy perused at great cost by telecoms equipment vendors such as Nortel was the acquisition of portfolios of internet

[20] ILOG productised technology originally developed by Nortel and adopted as industry GUI standard TMF046 and ITU-T Z.361 [Accessed Sept 2007] http://www.ilog.com/products/jtgo/

devices[21]. This was matched by the acquisition of portfolios of OSS products in order to provide a comprehensive managed internet offering for the service providers[22,23]. This strategy was predicated on the fact that while network equipment could (probably) interoperate at the transport and control layers, there was no comparable interoperability in the network management plane. The acquired internet devices had originally been developed for enterprise applications and the developers had not put the same effort into standardising the FCAPS management protocols and information models as had been put into traditional telecoms devices. There was little management commonality between IP equipment vendors and it was thought that acquisition of OSS products could add value by simplifying the integration process for a service provider who was purchasing a managed network from a single vendor - thus increasing that vendors equipment sales.

When the bubble burst, many OSS acquisitions were divested[24,25] and the industry swung away from a portfolio approach towards individual best of breed OSS products provided by a relatively small number of OSS ISV's[26], each with a relatively low market capitalisation (although it should be noted that Cisco has recently re-entered the market for acquiring OSS ISV's[27]). The equipment vendors, no longer seeing OSS as a competitive advantage also saw distinct disadvantages in not being able to integrate new equipment types with potential customers' OSS systems quickly. This has been a driver for developing common network interfaces such as MTOSI (*TMForum 2005e*) which has the support of major equipment vendors such as Cisco and Nortel.

Finally it should be mentioned than it appears to be becoming an increasingly common practice for equipment vendors to imbed open source into their systems. Alcatel has reported on their approach to using open source in development (*Ruffin, M. & Ebert, C. 2004*). Lucent has reported using open source development techniques within their company as 'corporate source' to facilitate rapid global product development of a SIP server (*Gurbani, V. K., Garvert, A., & Herbsleb, J. D. 2006*). Nortel confirmed that they had used JBoss in a corporate source development model and imbedded the open source DMTF CIM implementation OpenPegasus

---

[21] 15/6/1998 Nortel to buy Bay Networks for $9.1 billion http://www.networkworld.com/news/0615nortel.html [Accessed Sept 2007]

[22] October 18, 1999 Nortel Networks to Acquire Clarify for US$2.1 Billion, Establishes First Mover Advantage in Creating Second Wave of eBusiness [Accessed Sept 2007] http://www.nortel.com/corporate/news/newsreleases/1999d/10_18_9999661_clarify.html

[23] April 19, 2000 Nortel Networks to Acquire Architel for US$395 Million to Speed the Delivery of Next Generation Internet Services [Accessed Sept 2007] http://www.nortel.com/corporate/news/newsreleases/2000a/04_19_0000239_brookline.html

[24] October 2, 2001 Nortel Networks To Sell Clarify Assets To Amdocs Ltd for US$200 Million In Cash http://www.nortel.com/corporate/news/newsreleases/2001d/10_02_01_amdocs.html [Accessed Sept 2007]

[25] 1/28/2002 Nortel sells remainder of its OSS (Architel ASAP service activation system, Objectel inventory system, service-level management solutions and billing mediation software) assets to MetaSolv for $35 million. [Accessed Sept 2007] http://www.metasolv.com/MSLV/CDA/General/ArticleDetail.aspx?id=56

[26] To illustrate the size of the OSS industry, The Telemanagement World exhibition in May 2007 had only 116 exhibitors and not all were ISV's.

[27] Press Release: July 26, 2005 Cisco Systems to Acquire Sheer Networks for approximately $97 million [Accessed Sept 2007] http://newsroom.cisco.com/dlls/2005/corp_072605.html

(*TheOpenGroup 2007a*) in one product[28]. Furthermore a Nortel senior architect has written a book which describes practically how to use OpenPegasus and the DMTF CIM (*Hobbs, C. 2004*).

3.4.3.2 OSS Independent Software Vendors (ISV's)

OSS Independent Software Vendors (ISV's) (e.g. Cramer, HP, Agilent, Metasolve etc) provide OSS capabilities which are not tied to any one equipment vendor. ISV's are motivated to maximise the value of their product to their customers. OSS development is seen as a differentiator.

(*Economides, N. 1996*) notes that, intuitively, a firm benefits from a move to compatibility if (i) the marginal externality is strong; (ii) it joins a large coalition; and (iii) it does not thereby increase competition to a significant degree by its action. In a market of individual 'best of breed' OSS components, network externalities are very strong since each OSS component relies on components from other OSS manufacturers in order to function. There is a strong incentive to work on easily implemented compatibility standards and this was a driver behind the partners in the original OSS/J consortium (*TMForum 2004d*) specifying java interfaces in order to fast track a minimal level of NGOSS-like integration using J2EE and Web Service enterprise design patterns.

More recently, the pendulum has been swinging back towards portfolio OSS solutions with for instance the acquisition of Micromuse by IBM[29], Cramer by Amdocs[30] and Metasolve by Oracle[31]. It will be interesting to see whether the instinct will be to create monopoly portfolios of integrated products, or continue to develop public compatibility standards which allow substitutes.

Unlike the element managers from the equipment vendors, COTS OSS products are usually highly customisable. This is partly to allow adaptation to the more rigid element managers but it also reflects the fact that the business model lies somewhat between a rigid product cycle and a custom development. Often sales to large customers of COTS OSS products will require a degree of custom development or the contracted promise of a feature in a later release. In addition around 25% of the revenue from a large OSS COTS product sale can be consultancy and configuration for the environment.

The network relationships between OSS applications can be characterised using the emerging Telecoms Application Map (TAM) (*TMForum 2006a*). The TAM seeks to provide an industry agreed taxonomy for the domain specific functionality provided by OSS COTS products.

---

[28] Private Open OSS discussion with Nortel in Ottawa 12/4/2006

[29] IBM acquires Micromuse on 21 December 2005. USD$865 Million Acquisition follows the growing trend of acquiring partners [Accessed Sept 2007] http://research.analysys.com/default.asp?Mode=article&iLeftArticle=2046&m=&n

[30] August 16th, 2006 Amdocs Completes Acquisition of Cramer Systems for approx. $375 million [Accessed Sept 2007] http://www.ossnewsreview.com/telecom-oss/amdocs-completes-acquisition-of-cramer-systems/

[31] October 23rd, 2006 Oracle Corporation has agreed to acquire MetaSolv for $219.2 million http://www.ossnewsreview.com/telecom-oss/oracle-acquires-metasolv/ [Accessed Sept 2007]

Where previously OSS ISV's were obliged to market the business functions served by their systems according to the processes in the eTOM (*TMForum 2007b*), ISV's are beginning to produce marketing material describing their products according to the taxonomy in the TAM. However although the taxonomy helps in identifying areas where complementarities and substitutes could exist, it is only the first stage towards making OSS products compatible in a 'plug and play' sense.

As with any COTS product, the challenge facing an OSS ISV is to package a coherent set of features addressing a specific problem domain, which represents a value proposition attractive enough to command a high price point. For this reason, it is not always in the vendors (commercial) interests to allow their product to be picked apart by prospective customers. Real OSS products tend to blur the boundaries of the TAM functions, and often vendors aggregate more than one capability in a product in a way which prevents easy partitioning of functionality.

3.4.3.3 Middleware Component Vendors

Middleware Component Vendors provide components which broadly fall under the architectural categories listed in Table 15 as Support, User Interface, Middleware and Operating System and which are leveraged by the Equipment Vendors and ISV's in developing their Core OSS products but may also be used directly by Service Providers and Integrators operating in a 'component assembler' role. These products benefit from the economies of scale which come from being made available across the global software industry and not just confined to OSS.

Some vendors such as Sun, BEA, IBM, Microsoft or Oracle market middleware capabilities directly to the Service Providers, often in the form of a site-wide licence. In the case of Sun or IBM, the objective might also be to leverage server hardware sales. These players want to maximise the usage (and dependency) which the customers derive from their product. OSS is seen as a channel to market.

An emerging trend which must be causing the pure OSS COTS vendors some concern is the realisation that it may well be possible to build useful NGOSS systems using more generic middleware components. This has for instance been demonstrated using off the shelf Microsoft e-business components by Korea Telecom (*Cheol-Seong, K., Joong-Goo, S., Dong-Sik, Y., & Kyung-Sup, S. 2004*), (*Cheol-Seong, K., Ok, K., Hong, D. W., Chung, B., & Bang, Y.-H. 2006*). This suggests that functionalities which traditionally have been packaged by OSS COTS vendors are becoming commoditised and that COTS vendors should be thinking about how they can move up the value chain. This may well change the competitive landscape if the OSS vendors think they need to cooperate more in order to counter this perceived threat.

The generic middleware vendors (with the exception of Microsoft who claim most customers don't want access to the code (*Matusow, J. 2005*)) appear to have been the most active in

perusing an open source strategy. An increasing number of companies are open sourcing previously COTS components (e.g. Sybase (*Marson, I. 2004*)).

Sun Microsystems is a major player in the telecoms space which is aggressively perusing an open source strategy. In addition to the NetBeans IDE (*NetBeans 2007*) and OpenOffice (*OpenOffice 2007*), Sun has open sourced Solaris (*SUN Microsystems 2005a*) and Java (*SUN Microsystems 2006a*). Sun initiated the OSS/J program which actively encourages open source contributions (*OSS/J 2007c*). They are also sponsoring the OpenESB project (*OpenESB 2007*) which they actively promoted to the telecoms community at Telemanagement World Nice 2007.

Hewlett Packard also has a much publicised open source strategy called 'Progressive open source' (*Dinkelacker, J., Garg Pankaj K., Miller, R., & Nelson, D. 2002*) and the director of their Linux program has written extensively on the business and economics of open source (*Fink, M. 2002*).

IBM's most visible open source contribution has been the Eclipse IDE (*Eclipse 2007a*) but is also making open source a key component of its business strategy (*Worthington, D. 2005*), (*IBM 2007a*). IBM is also sponsoring the eclipse based network management project COSMOS[32]

3.4.3.4 Specialist OSS Middleware Component Vendors

Fourthly some specialist vendors provide more specialised OSS middleware components such as GUI widgets and rules engines (ILOG) or object data bases (Versant) which are used as components within core OSS systems. A recent entry into this market would be Nokia's OpenEMS (*Nokia 2007*) platform which provides generic element management capabilities which other equipment vendors could use to build their own Element Management System (EMS). This itself leverages the open source Netbeans platform for the GUI.

The OSS specialist middleware vendors also appear willing to leveraging open source to construct their COTS products and one long time player IONA has published their open source strategy and is delivering an open source enterprise service bus[33].

OSS tool vendors would also fall into this category. For example, Telelogic/Popkin[34] providing eTOM based business process modelling tools or TigerStripe[35] providing OSS/J interface plug-in's for Eclipse.

3.4.4 Component Assemblers

Component Assemblers locate suitable components and assemble them in integrated application systems that satisfy customer requirements (*Vitharana, P. 2003*). This category

---

[32] COSMOS announced in October, 2006 (COmmunity-driven Systems Management in Open Source) open source implementation of Service Management Language SML http://www.eclipse.org/cosmos/faq/ [Accessed Sept 2007]

[33] http://www.iona.com/solutions/opensource/ IONA Open Source Solutions [Accessed Sept 2007]

[34] http://www.telelogic.com/ [Accessed Sept 2007]

would cover most integrators in the OSS space including Invocom (the sponsor of this research) and Indian integrator Cognizant (who participated in the catalyst). However many of the larger service providers also do this work internally.

In order to understand the issues in deploying open source COTS in a large operator it is necessary to consider where these players fit within the overall process of COTS system deployment in a service provider. (*Torchiano, M. & Morisio, M. 2004*) provides a simplified view of the roles of COTS Vendors, System Integrators, Customers and end Users in COTS based development as illustrated in Figure 7. A COTS vendor creates a product which is chosen and adapted by a system integrator to meet the needs of the customer and ultimately the end user.

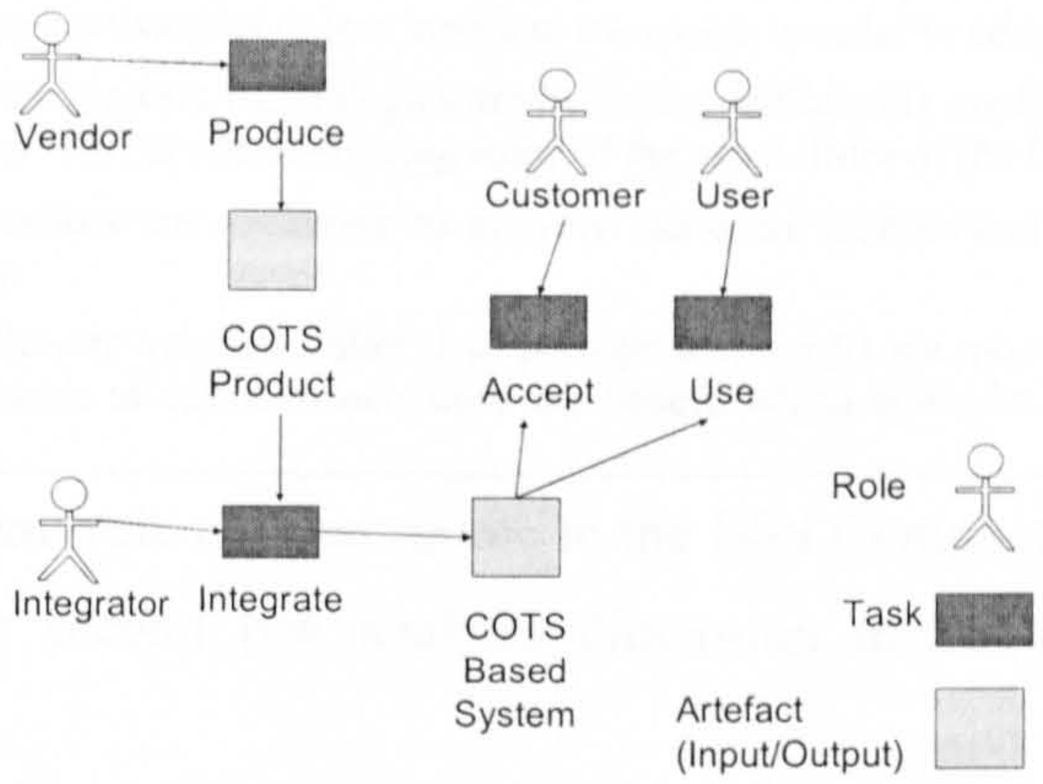


Figure 7 Simplified view of COTS based development
(after Torchiano, M. et al. 2004)

However COTS Systems introduction is a significantly more complicated process in a large service provider. The complexity lies in two dimensions; firstly the problem of technical fit (which is discussed as part of the development approach in Section 3.5) and secondly the problem of process fit discussed below.

At a generic level the problem of process fit can be understood by considering the processes by which telecoms systems are brought into operation in the eTOM process model GB921 (*TMForum 2007b*). The eTOM is the only process model which provides a classification and relationship between all the business processes present in a typical large service provider. It is not specifically geared to software systems but concerns all of the platforms which a service provider uses to run its business. An analysis of the generic eTOM processes affected by the introduction of open source is given in Appendix C. At a detailed level, each operator will have different processes for COTS introduction, perhaps drawing on best practice as documented by ITIL (*ITIL 2007*) or SEI (*Brownsword, L. et al. 2000*) or even more formal processes like those described by (*Ochs, M. A., Pfahl, D., & Chrobok-Diening, G. 2001*).

[35] http://www.tigerstripesoftware.com/ Tigerstripe Software [Accessed Sept 2007]

3.4.5 Summarising the OSS Component Market Place

This section can be summarised with the following data points in Table 16;

Table 16 Summarising the OSS Component Market Place
1. The telecoms industry exhibits strong network effects which motivates otherwise competitors to cooperate in order to grow the market. With increasing interoperation, there is stronger motivation to cooperate at the OSS level.
2. Rapid innovation in hosted applications above the 'TCP/IP partition' is facilitated in many cases using open source.
3. Although much R&D has moved from the service providers to specialist technology vendors, service innovation is distributed throughout the service providers organisations and the residual R&D activities are concerned with learning how to use (not develop) new technology.
4. Despite the growth of the COTS market, globally service providers still do a lot of internal development. If open source projects representing common interests were in existence, they could draw on these resources as users.
5. Many large service providers have formidable problems introducing open source due to the complexity of changing their processes. Smaller providers, niche players and players in emerging markets have stronger motivations or less inertia to overcome in order to adopt open source.
6. Generic e-business middleware technologies are becoming sufficiently sophisticated to be configured for NGOSS functions. This is commoditising some of the capabilities of the OSS COTS vendors.
7. Major equipment vendors are beginning to actively use open source components and development techniques internally.
8. Major generic middleware vendors (other than perhaps Microsoft) are moving towards open source. The OSS industry needs to adjust to maximise the benefit which could be derived from these open source technologies.

Having analysed the current role of open source in the OSS Component Marketplace the next section will look at the second risk analysis dimension in Table 10, the Development Approach.

3.5 Development Approach

The discussion of the role of open source as an OSS development approach naturally divides into short term (tactical) and long term (strategic) uncertainty management considerations.

In an exploratory study, (*Torchiano, M. et al. 2004*) interviewed seven small and medium sized software companies in Norway and Italy about their experiences of COTS based development and summarized their results in six theses (T1-T6 below) which help with framing the tactical and strategic questions addressed in the following sections.

3.5.1 Industry Strategy

3.5.1.1 Strategic Questions Concerning OSS Open Source

The strategic questions are concerned with understanding what are the long term risks and opportunities associated with developing a market for OSS components, i.e. how should the industry steward the long term development, availability, evolution and uptake of the most useful OSS components? These are questions of interest to CTO's, Architects Product Managers and those setting the direction of industry standards.

(*Torchiano, M. et al. 2004*) found that developers seldom use formal selection procedures for COTS products[T4]. Familiarity with either the product or the generic architecture was the leading factor in selection. Furthermore, developers considered architecture more important

than requirements for product selection[T5] and Integrators tended to influence the vendor on product evolution whenever possible[T6]. This leads the first strategic question;

- Strategic Question 1: Could open source help the OSS industry ensure that the generic NGOSS architecture and NGOSS components are aligned with users requirements, easy for developers to comprehend, and malleable to direct feedback from integrators' requirements and experiences?

(*Torchiano, M. et al. 2004*) found that most COTS integration problems resulted from lack of compliance with standards and that architectural mismatches constituted a secondary issue[T2]. In other words if a component claimed compliance to a standard interface, it was often the case that this was incorrectly implemented due to the complexity of the interface and un-familiarity on the part of the developer. (*Torchiano, M. et al. 2004*) argue that this points to the need for rigorous COTS compliance testing (which the TMForum is now planning as part of the Prosspero program), however the problem might also be eased by a compliant open source implementation. This leads to the second strategic question;

- Strategic Question 2: Would an open source commons of OSS reference implementation code, maintained by the standards developers, de-risk projects if the code was made available for ISV's to incorporate into their COTS components?

3.5.1.2 Sources of Strategic Uncertainty in the OSS Component Marketplace

Building systems using COTS components potentially offers a number of advantages. The desired functionality can be accessed immediately, obtained at a significantly lower price (than bespoke development) and developed by someone expert in that functionality (*Voas, J. 1998*).

This experience embedded in a product design can be particularly important for a new service provider. (*Avison, D., Gregor, S., & Wilson, D. 2006*) note that an Australian telco, One.tel went bankrupt in 2001 partly because their billing system was created and managed by a team who had little experience in developing such systems.

Successfully designing and sustaining a system using COTS based components requires a much wider set of processes and relationships than bespoke development. The SEI define COTS based system development as '*occurring through the simultaneous definition and trade-off of the COTS marketplace, system architecture, and system requirements*' (*Brownsword, L. et al. 2000*). They note that there is frequent, continuous change in COTS products and the marketplace and this is a significant risk which needs to be managed.

(*Boehm, B. W., Port, D., Yang, Y., & Bhuta, J. 2003*) have identified four categories of COTS based system development as listed in Table 17. Although historically service providers' OSS systems were developed internally using 'Non-COTS intensive development', most projects in tier 1 providers (Table 12, Section 3.4.2) would now be categorised as Glue-Code COTS Based Applications (GCBA), if for no other reason than the new system has to fit with a number of legacy systems and processes. However a 2nd or 3rd tier service provider (e.g. a metro area cable operator or a large enterprise), with relatively un-sophisticated integration requirements might purchase a small selection of COTS systems (ACBA) and perhaps do some minor tailoring (TCBA).

If bespoke development is required, most service providers prefer to contract the COTS vendors to undertake it as an extension to their product rather than do it in house. It is interesting to contrast this with players above the TCP/IP demarcation like Google whose differentiator is bespoke search engine algorithms, which they implement in house on top of an open source COTS infrastructure in the form of Non-COTS intensive projects.

Table 17 Categorisation of COTS based projects
(after Boehm, B. et al. 2003)

1. Assessment COTS Based Application (ACBA)	Requirements covered without much modification (or tailoring) by a single COTS package or through the simple integration of several. Main effort: Identifying candidate COTS products whose existing capabilities directly address a desired operational concept.
2. Tailoring COTS Based Applications (TCBA)	Requirements are covered by a single (or very few) COTS package(s). Main effort: Adapting a COTS framework (data fill & configuration) whose existing general capabilities can be customised (i.e. tailored) to the majority of requirements.
3. The Glue-Code COTS Based Application (GCBA)	Non-Trivial requirements covered by multiple COTS packages and components. Main effort: Identify multiple COTS packages to address subsets of the required system capabilities and integrate them to develop the system. Typically requires significant glue code design and development in order to integrate such packages.
4. Non-COTS intensive (bespoke development)	The non trivial requirements require custom design and development but the system leverages COTS components. Typically there are significant "Buy versus Build" decisions. Critical factors are: availability of COTS products, schedule limitations, budget limitations, and desired current and future capabilities. Main Effort: custom design and development.

3.5.1.3 NGOSS as a means for Strategic GCBA Risk Reduction

NGOSS is intended to simplify and reduce the risks associated with GCBA development by standardising the key architectural principles embodied in COTS OSS products.

Drawing on (*Szyperski, C. et al. 2003*) and (*Wallnau, K., Bachmann, F., Bass, L., Buhman, C., Comella-Dorda, S., Long, F. et al. 2000*), an NGOSS component is defined as a 'contractually specified unit of independent deployment subject to third party composition and conformant to a component model' (*TMForum 2006c p24*).

The NGOSS Technology Neutral Architecture (TMF053) (*TMForum 2006c p23 ff*) envisages a distributed system providing Framework Services and Business Aware Services which are contractually specified and deployed via components. Framework Services provide non-business orientated services to the system such as naming and directory services, messaging services and policy distribution services. Business Aware Services provide application level functionality that supports the implementation of the OSS business process; customer record management, SLA management, QoS management etc.

NGOSS services are defined in terms of the contracts which NGOSS components expose. At the highest level NGOSS defines a contract in accordance with (*Szyperski, C. et al. 2003*) as 'a specification attached to an interface that mutually binds the clients and providers (implementers) of that interface'. Contracts can cover functional and non functional aspects. Functional aspects include the syntax and semantics of the interface. Non functional aspects include quality of service guarantees (*TMForum 2006c p7*).

Current OSS GCBA application development is dramatically complicated by the fact that most contemporary OSS systems already incorporate proprietary implementations of framework services and business services which are so intertwined that external identification and separation is no easy task.

Introducing NGOSS to the industry may thus be thought of as a re-factoring exercise where ISV's are encouraged to pick apart contemporary COTS OSS systems into their constituent components so they can be flexibly re-assembled according to the service providers' wishes. Alas, this unbundling of functionality may not be perceived to be in the current COTS ISV's commercial interests, a subject discussed in Section 3.4.

The interim solution often adopted by service providers is to add another set of framework services on top to integrate the disparate applications. This adds cost and complexity to the design (*TMForum 2006c p24*). Examples of this would be BT's use of J2EE based 'middleware hubs' (*Reeve, M. H., Bilton, C., Homes, P. E., & Bross, M. 2005*), (*Strang, C. J. 2005*), or Vodafone's adoption of OSS/J into their trouble ticket mediation architecture (*Buschmann, A. & Hoeckelmann, G. 2007*). If NGOSS components became available, the integration task would in theory become much more of a plug-and-play exercise.

The industry migration path towards NGOSS thus appears to be;

a) An initial introduction of an overlay NGOSS derived 'glue ware' to solve the problem of integrating disparate applications by standardising on a common communications environment (e.g. OSS/J or MTOSI), process model (eTOM) and information model (SID).
b) The gradual introduction of native NGOSS components as they become available using mediation interfaces between legacy non-NGOSS components and the NGOSS environment

The ultimate goal of NGOSS is the fostering of a telecoms OSS component market place. This was primarily conceived of in terms of components delivered as COTS products but does not preclude the use of open source components. (*Brownsword, L. et al. 2000*) outline several generic sources of risk in pure COTS development. It can be argued that the OSS stakeholders seek to manage many of these risks through their activities in the TMForum. Table 18 lists some areas where TM Forum activities already manage the risk and where open source activities could further reduce the risk of COTS development.

We can observe that the TMForum is trying to reduce the uncertainty of COTS based OSS development through its standardisation activities and through limited proofs of concept orchestrated through the Catalyst programs. However this is not leading to the radical introduction of NGOSS Components which would deliver the common Framework Services which the ISV's will need to re-factor the Business Aware Services out of their products.

Given the somewhat chequered history of OSS standardisation discussed in Section 1.4 it is unlikely that a paper specification will be enough to move this situation forwards. Therefore one can argue that perhaps the most effective means to begin to introduce true NGOSS components will be to invite all of the stakeholders and industry experts to collaborate in initial open source reference implementations of the most commonly needed Framework Services on which differentiated products can be built.

Our question then becomes,

- Strategic Question 3: Could the strategic use of open source further reduce the risk of the standardisation process failing to deliver useful standards (i.e. address the problems described in Chapter 1)?

Table 18 Managing frequent, continuous change in OSS COTS marketplace

Source of COTS Market Risk (*Brownsword, L. et al. 2000*)	TMForum / NGOSS risk reduction	Open Source risk reduction
1. The marketplace, not the needs of any particular system, drives COTS products.	• TMForum provides a 'controlled marketplace'. • Catalyst program proves interoperability	• Catalysts built using open source OSS components could be replicated in local labs and build a common understanding across the industry.
2. Products have built-in assumptions about how they will be used; these might not match the system users' processes, resulting in clashes.	• eTOM standard seeks to standardise telecoms processes and use cases.	• Open source OSS components would allow each service provider to test the solution using their local use cases and feed back improvements
3. Products are built on architectural assumptions that can vary across system components and could conflict with an evolving system architecture.	• NGOSS seeks to standardise key architectural principles. • SID seeks to standardise information models and system interfaces.	• Open source OSS components providing NGOSS Framework Services would provide a common base for the industry to begin opening up existing applications.
4. COTS products will have interdependencies.	• TAM seeks to characterise and partition COTS functionality. • NGOSS Contracts concisely define the offered functionality and interdependencies. • Prosspero seeks to commercialise interfaces in a tested framework.	• Open source OSS components providing NGOSS Framework Services would go beyond Prosspero and provide a common layer for resolving inter component dependencies.
5. Licensing and data rights will affect cost, architecture, and user processes.	n/a	• Open Source modifies licence restrictions on architecture.
6. Projects have limited control over a COTS products release frequency or content.	n/a	• Open Source gives visibility to release plans and content - and some control.
7. Projects have limited visibility into COTS products' source code and behaviour.	n/a	• Open Source allows in depth understanding of product architecture and behaviour.

3.5.2 Tactics for a successful OSS project

3.5.2.1 Tactical Questions concerning OSS Open Source

The tactical questions are concerned with understanding how to manage the short term uncertainties associated with using COTS or Open Source to deliver a project. These are questions of interest to project managers and developers charged with delivering a specific OSS project for a particular company.

(*Torchiano, M. et al. 2004*) found that projects often use open source as closed source[T1]. This leads to the first tactical question;

- Tactical Question 1: If an OSS project only intends to use open source components without modification in the same way as it would closed source components, would the fact that the code is visible allow the developers to better assess its functionality, detect and fix bugs and reduce the risk to the overall OSS project?

(*Torchiano, M. et al. 2004*) found that projects using COTS components mostly used development to provide additional functionalities not just glue ware[T3]. This leads to the second tactical question;

- Tactical Question 2: If it is generally the case that real OSS projects often need to extend the existing functionality of COTS components to meet their needs, would it be more cost effective to have access to the source code of the COTS components and work out how do it yourself than to pay a COTS vendor to do it when they don't really understand your context?

3.5.2.2 Sources of Tactical Uncertainty in Open Source or COTS Based Development

When considering the risks surrounding the development of NGOSS applications using COTS or open source components, it is important to consider the granularity of the components involved. Bloem's consideration of COTS application development (*Boehm, B. W. et al. 2003*) has examined the issues around the selection and integration of larger low granularity COTS applications. (*Li, J., Bjørnson, F. O., Conradi, R., & Kampenes, V. B. 2004*) have focused on the issues of using highly granular COTS software utility components in the form of COM, EJB, C++ or Java libraries.

By definition the granularity of NGOSS components must fall between those considered by (*Boehm, B. W. et al. 2003*) and (*Li, J. et al. 2004*). Furthermore, the migration path to NGOSS discussed in Section 3.5.1.3 suggests that the industry will use wrapper technology around existing OSS applications as the first step towards NGOSS. This wrapper technology will inevitably be constructed using high granularity utility or middleware components and libraries, many of which already have their open source equivalents. Therefore in considering the risks and opportunities of using open source for NGOSS systems it is not unreasonable to look at others' experiences of integrating both software components and full applications.

Following a review of relevant literature, (*Li, J., Reidar, C., Slyngstad, O. P. N., Bunse, C., Khan, U., Torchiano, M. et al. 2007*) have identified a set of 15 risks commonly associated with projects using off the shelf software components to develop larger applications (Table 20). They used these risk categories to construct a comparative survey into the differences in risk experienced between projects building applications using open source components and those using COTS software components as scoped according to the definition in (*Li, J. et al. 2004*). This survey covered 115 projects (47 from Norway, 25 from Italy, and 43 from Germany) from a cross section of industries and included a small number of telecoms projects.

(*Li, J. et al. 2007*) found that the risk profile of projects using Open Source components in development was broadly similar to using COTS components, a fact supporting the conclusions of (*Giacomo, P. D. 2005*), that open source components are not completely different from COTS components. However (*Li, J. et al. 2007*) found that COTS and open source components risk profiles differed in the respects listed in Table 19. This leads to the third tactical question;

- Tactical Question 3: What is the sensitivity of a particular OSS project to the risks of Estimation, Adaptation, Security and Support and how will the choice of COTS or OSS component effect the projects risk profile.

Table 19 Differences between COTS and open source components risk profiles (Li, J. et al. 2007)	
1. Estimation:	During the project estimation phase, COTS users experienced a higher risk of poor estimation possibly because they could not see the source code in order to fully understand if the component was suitable (R2)
2. Adaptation:	COTS users found it more difficult to adapt their components to meet changing requirements (R5), again, possibly because they could not modify the components at a source code level.
3. Security:	COTS users found it difficult to control the components effect on security (R8) again possibly because they couldn't see inside the component to identify security holes
4. Support:	Open Source users had a higher risk in quantifying the open source projects support reputation when there was no support contract in place. However in practice, teams using open source were not less satisfied with the technical support than COTS users but (*Li, J. et al. 2007*) note that this may have reflected the respective development teams confidence in digging into the open source components for themselves.

Table 20 Project Risks Associated with Using Off The Shelf SW Components (after Li, J. et al. 2007)

Phase	ID	Possible risks	difference between COTS/open source
Project plan phase	R1	The project was delivered long after schedule (*Rose, L. C. 2003*)	same
	R2	Effort to select OTS components was not satisfactorily estimated (*Boehm, B. W. et al. 2003*)	open source better
	R3	Effort to integrate OTS components was not satisfactorily estimated (*Rose, L. C. 2003*)	open source better
Requirements Phase	R4	Requirement were changed a lot (*Boehm, B. W. et al. 2003*)	same
	R5	OTS components could not be sufficiently adapted to changing requirements (*Boehm, B. W. et al. 2003*)	open source better
	R6	It is not possible to (re)negotiate requirements with the customer, if OTS components could not satisfy all requirements (*Li, J. et al. 2004*)	same
Component integration phase	R7	OTS components negatively affected system reliability (*Ruffin, M. et al. 2004*), (*Vitharana, P. 2003*)	same
	R8	OTS components negatively affected system security (*Ruffin, M. et al. 2004*) ,(*Boehm, B. W. et al. 2003*) (*Vitharana, P. 2003*)	open source better
	R9	OTS components negatively affected system performance (*Vitharana, P. 2003*)	same
	R10	OTS components were not satisfactorily compatible with the production environment when the system was deployed (*Vitharana, P. 2003*)	same
System maintenance and evolution	R11	It was difficult to identify whether defects were inside or outside the OTS components (*Ruffin, M. et al. 2004*)	same
	R12	It was difficult to plan system maintenance, e.g. because different OTS components had asynchronous release cycles (*Rose, L. C. 2003*)	same
	R13	It was difficult to update the system with the last OTS component version (*Rose, L. C. 2003*)	same
Provider relationship management	R14	Provider did not provide enough technical support/ training (*Ruffin, M. et al. 2004*) , (*Rose, L. C. 2003*)	same
	R15	Information on the reputation and technical support ability of provider were inadequate. (*Ruffin, M. et al. 2004*) ,(*Rose, L. C. 2003*)	COTS better

3.6 Collaborative Innovation within the TMForum

We now turn to the question of whether open source could contribute to more rapid collaborative innovation between firms in the TMForum?

3.6.1 The need to integrate R&D and Marketing in New Product Introduction

Unlike mass market products, most OSS systems sold to service providers require a complex sales process. A complex sale is one where a number of people must give their approval or input before the buying decision is made (*Miller, R. B., Heiman, S. E., & Tuleja, T. 2005 p28*). Strategic Selling is the process by which a vendor's account team cultivates a strategic set of relationships in order to influence the decision making process in large purchases. These relationships need to be fostered both vertically and horizontally between the peer business, technical and user functions in both organisations.

Similar relationships are needed in order to find 'lead customers' in order to help scope the requirements for new products. One of the challenges in running a successful New Product Introduction (NPI) process is in closing the communications gaps between functions both within and between the customer and supplier organisations as illustrated in Figure 8. Of course this is greatly simplified; the true complexity of the functional departments needing influenced is discussed in terms of the eTOM in Appendix C.

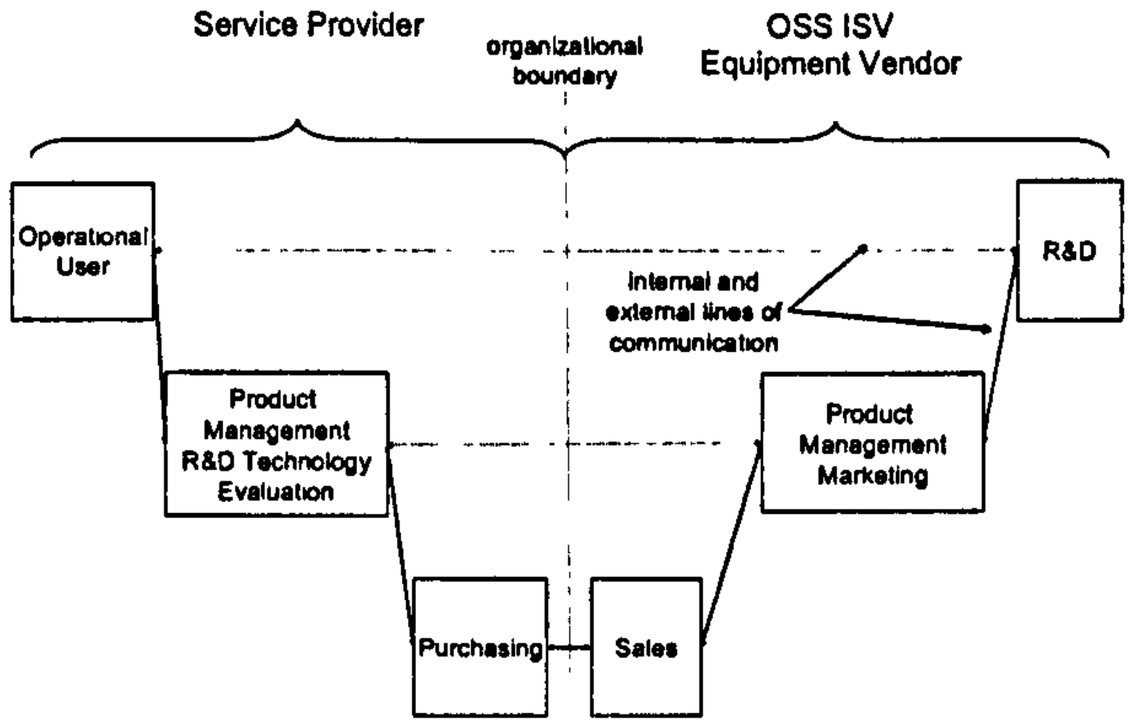


Figure 8 Communications gaps between functions in Vendors and Customers

Realistically, OSS vendors have limited resources to engage with customers and cannot be certain they are targeting the right people, particularly given the organisational volatility of most service providers. A significant potential benefit of open source is its ability to establish trusted relationships which bypass the organisational gatekeepers (sales and purchasing) and allow end users and technical evaluators to directly engage with, evaluate and influence the technology in an extended pre-sales discussion. Thus open source can be used an important part of the marketing and sales strategy. One objective for this research is to investigate how in practice this might work in the OSS market place.

(*Griffin, A. & Hauser, J. R. 1996*) list a number of criteria for measuring successful NPI processes including measures of the rate of generation of new product ideas, the proportion of ideas successfully brought to market and the extent to which products capture market share and generate revenue and profits. The traditional roles of R&D and Marketing in the NPI process are summarised in Table 21. A key purpose of both R&D and marketing is to reduce business risk due to market or technology uncertainty throughout the NPI process.

Table 21 Summary of marketing and R&D roles (after Griffin, A. & Hauser, J. R. 1996)	
Function	Role activities Include:
Marketing (outbound)	Selling product, talking to customers and communicating product benefits, creating accurate product literature and selecting advertising claims and trouble shooting customer problems.
Marketing (inbound)	Finding and assessing new applications for products and technologies, providing information on customer needs and participating in decisions on product positioning and feature delivery.
R&D	Maintaining and developing technology, establishing long term research directions, keeping abreast of competitive technology and identifying and fixing design flaws for future releases.

(*Griffin, A. et al. 1996*) summarise the literature on NPI by suggesting that while much evidence indicates that project success requires careful communication and cooperation between Marketing and R&D, empirical research also indicates that disharmony between marketing and R&D is the rule rather than the exception in many firms. (*Pfleging, B. & Zetlin, M. 2006*) have called this the 'geek gap' between technology and business professionals.

In the Telecoms industry this problem has been identified both by service providers and their suppliers. (*Lapierre, B. H. 1996*) did empirical research on new product introduction in a 'large Canadian telecommunications provider' (possibly from the description Bell Canada), which while outsourcing much R&D to suppliers also retained an 'applied R&D' function within the company to assist in activities associated with the deployment of network infrastructure. Managers reported a number of areas where the engineering and marketing functions did not communicate effectively. Similarly, researchers at Bell Northern Research (now Nortel), indicated the need to integrate R&D and marketing in manufacturing (*Seregelyi, E. V. & Kohut, I. A. 1991*).

(*Griffin, A. et al. 1996*) summarised the previous research, suggesting that there were numerous barriers to cooperation which tended to cause R&D and Marketing to drift apart. These included physical barriers (different locations), cultural thought worlds (nature of professional training), language (technical terms used) and differing organisational career paths (and hence resistance to organisational change).

Among the most frequently cited barriers to communication were the cultural differences arising from the different personality traits though to be attracted to R&D or Marketing. The early work of (*Saxberg, B. O. & Slochum, J. W. Jr. 1968*) proposed a list of Marketing and R&D personality stereotypes (Table 22). Even if these old stereotypes no longer accurately reflected the reality of the current population, (*Griffin, A. et al. 1996*) argued that the 'stereotype perception' could still be a formidable barrier to communication between groups. (*Griffin, A. et al. 1996*) went on to proposed a single model collating the various 'integrating mechanisms' which had been suggested in the literature for improving cooperation (Table 23).

Table 22 Summary of Marketing and R&D Stereotypes (after Griffin, A. & Hauser, J. R. 1996)		
	R&D Traits	Marketing Traits
Goals and Aspirations	• Knowledge as source of value to mankind • Research for research sake • Peer evaluation and recognition	• Organisational survival and growth • All activities relevant to firm's objectives • Organisational recognition
Needs	• Autonomy • Peer recognition and creative environment • Continuing education and development • Support for advancing knowledge in society	• Plans, procedures, policies and rules • Organisational recognition • Team work • Increased organisational status
Motivation	• Service to mankind • Publications • Professional recognition • Patents with name attached • Freedom to solve problems and advance knowledge	• Rewards and sanction system with advancement through organisation

Table 23 Integrating Mechanisms to improve R&D/Marketing cooperation (after Griffin, A. & Hauser, J. R. 1996)
1. Relocation or physical facilities re-design, 2. Personnel movement (e.g. Short term assignments), 3. Aligning incentives and rewards, 4. Adjusting the organisational structure (e.g. Cross functional project teams), 5. Encouraging informal social contact between the groups 6. Developing 'formal integrative managerial processes' to integrate the work of R&D and Marketing. Additional mechanism proposed by (Leenders, M. A. A. M. et al. 2002) 7. Strategic use of ICT

(*Griffin, A. et al. 1996*) noted that the literature broadly defined two forms of formal integrative managerial processes. In the less integrated form a 'Phase Review process' would ensure that uncertainty was reduced at each stage of the NPD process by each group performing tasks to a certain schedule. In the more integrated form a 'Stage Gate process' would synchronise marketing and R&D activities at review gates, before work could proceed to the next phase.

A number of studies have verified aspects of the model proposed by (*Griffin, A. et al. 1996*), however more recently (*Leenders, M. A. A. M. & Wierenga, B. 2002*) sought to quantify simultaneously the relative importance of all aspects of the model through empirical research using a survey of managers in 142 pharmaceutical companies. They consider the results to be generally applicable to any high tech industry. (*Leenders, M. A. A. M. et al. 2002*) argued that it is important to extend Griffin's model (1-6 in Table 23) to include the use of ICT as a 7th integrating mechanism. Perhaps unsurprisingly they found that better R&D and Marketing integration did improve new product performance and that all seven integrating mechanism had a positive effect on integration. However they also found that the use of Formal Integrative Managerial Processes (6) and ICT (7) had a direct effect on new product performance.

However although Formal Integrative Managerial Processes (6) made a significant contribution to bridging the gap between marketing and R&D, these processes also appeared to introduce substantial negative side effects on new product performance. (*Leenders, M. A. A. M. et al. 2002*) argued that this might be due to the formal process reducing employee initiatives and causing loss of flexibility. While formal processes were beneficial, companies

needed to ensure that they didn't de-motivate staff, quench the number of new ideas or generate too much paperwork.

The use of ICT (7) had a direct positive effect on new product performance. (*Leenders, M. A. A. M. et al. 2002*) suggest that ICT facilitates socialisation and externalisation processes that play an important role in the acquisition of tacit knowledge from customers and fellow researchers. ICT allows functional specialists to reconnect, not through physical space but as on-line communities and as this study demonstrated, also facilitated communication between functional areas.

If one considers open source development methods to be an extension of the role of ICT in the NPI process, it becomes an additional means of integrating communication between functional areas. One can conjecture that including an open source process within the NPI process could have a number of beneficial effects on new product development as listed in Table 24.

Table 24 Potential effect of Open Source on NPI process
1. The autonomy afforded to developers within a social context of an open source project could lead to more job satisfaction as it better matches their personality traits.
2. The reduction in formal process could lead to greater creativity.
3. The role of marketing changes from gatekeeper towards attracting and gauging interest from potential customers in joining a project as a pre sales activity.
4. Once a product is reaching the stage of needing industrialisation and support a more traditional NPI process would be engaged to make the product production ready.

The EU IST project ROCKET[36] has conducted research into the role of ICT in underpinning organisational knowledge management which they see as particularly relevant in knowledge intensive-processes such as product innovation. They acknowledge the contribution of (*Griffin, A. et al. 1996*) in identifying that communication is a fundamental problem in complex projects (*ROCKET 2003 p20*). Applying the work of (*Castells, M. 2000*), they foresee product innovation increasingly becoming a function of 'open-source' networks of cooperation where open-source networks are composed of teams of company employees and entrepreneurs within, as well as across, the formal boundaries of organisations. They suggest that innovation is becoming not something that is happening 'inside' firms but rather at the interfaces of firms with the market, regulatory and institutional environments within which firms operate (*ROCKET 2003 p27*). Clearly under this definition open source is seen as much more fundamental to future product development than just a programming model.

3.6.2 Role of the TMForum, Universities and other Non-Market Institutions

We have seen that products in the OSS space experience strong network effects in that the products rarely exist in isolation. One would therefore expect that successful introduction of an OSS product will require that complementary/compatible products and services are also in place, a process often called Market Adaptation. Research into the institutional domain of technology diffusion (*Reddy, N. M., Aram, J. D., & Lynn, L. H. 1991*) has further suggested

that companies often use non-market institutions to influence sources of technological complementaries. Non-market institutions include trade associations and universities as illustrated in Figure 9.

Companies participate in trade associations like the TMForum in order to adapt the market to their product and also to reduce the risk of new product development by agreeing compatibility standards such as NGOSS. Table 18 in Section 3.5.1.3 illustrated some of the COTS risk reduction facilitated through the activities of the TMForum.

Universities are also considered potentially influential non-market institutions. The TMForum has recognised this by creating a university liaison program in order to attract more research into the industry (*TMForum 2007e*). However the required time, resources and access to proprietary code, realistically prevents the average research student from addressing large scale OSS problems without having a support infrastructure in place before they start.

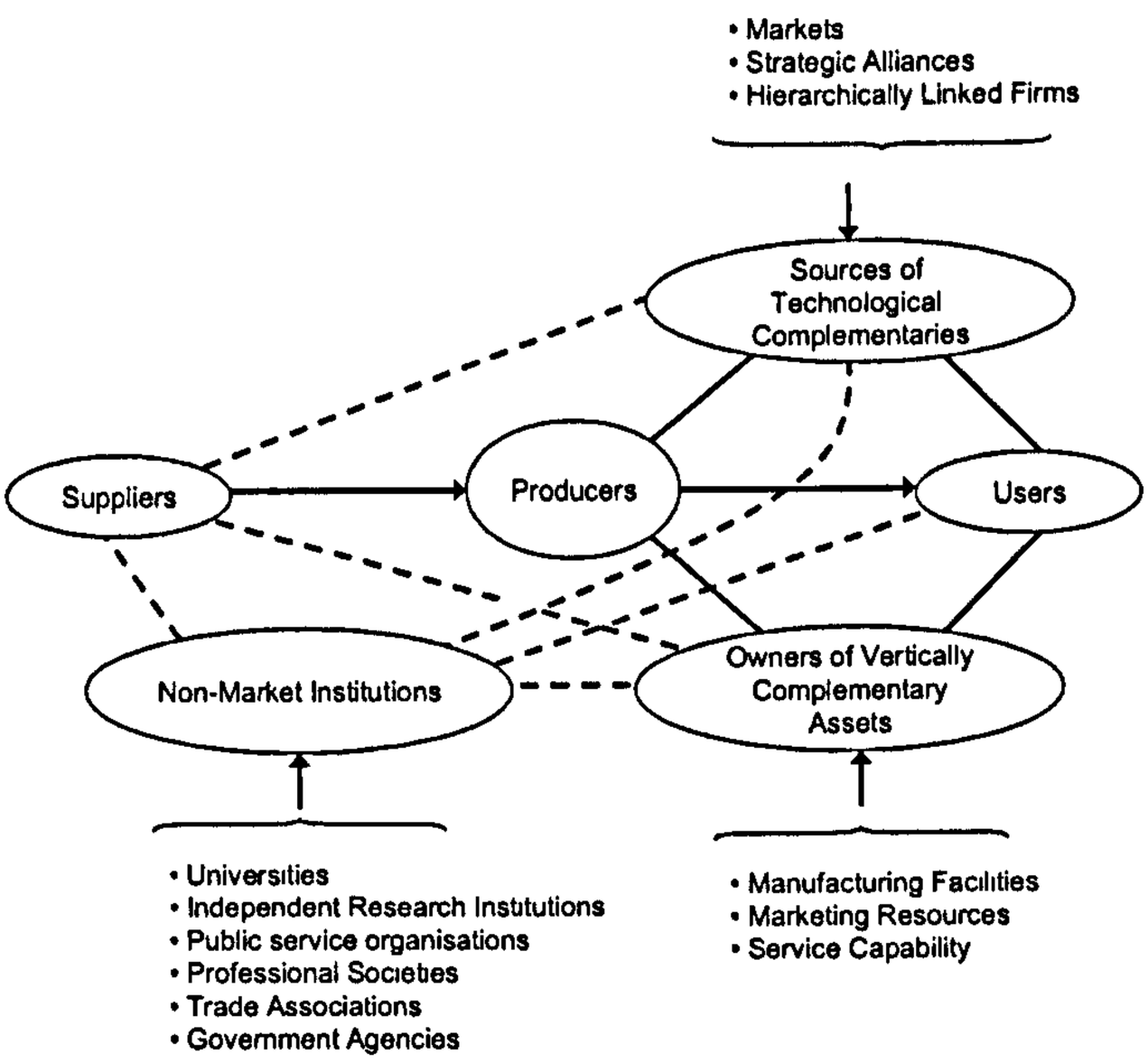


Figure 9 The Innovation Community
(after Reddy, N. M., Aram, J. D., & Lynn, L. H. 1991)

Indeed, this could be why much academic network management research has concentrated on enterprise solutions where resources are more easily available. The enterprise focused DMTF have encouraged the use of open source to both develop and promote standards which makes academic access easier (*DMTF 2004*). The DMTF has around fifty-eight academic partners in their academic partnership program while the Service provider focused TMForum has about five. The OSS industry needs to attract more research interest to redress the balance. Using open source to extend the Catalyst program into a 'living lab' which is available to academia would obviously make academic access easier.

In some senses, Open Source itself can be considered a non market institution. (*Gehring, R. A. 2006*) argues that it is not one single property of Open Source software such as, for example,

[36] ROCKET Roadmap to Communicating Knowledge Essential for the Industrial Environment" is an EC funded project that was approved in the last call of the fifth framework. Completed in 2003 http://rocket.vub.ac.be/ [Accessed Sept

the absence of licensing costs, that attracts industry leaders. Rather it is the institutional framework of Open Source that makes it an attractive alternative mode of organizing the production and distribution of software and software-related services.

The OSS industry is beginning to recognise the value of open development techniques in order to agree compatibility standards. OSS/J would be an example of a non market institution which has delivered a (partially) open source solution to developing OSS compatibility standards through the Java Community Process. The MTOSI reference implementation developed by CISCO would be another example[37].

Therefore another objective of this program is to test the following conjecture;

> Conjecture: That the OSS industry might have a great deal to gain if it could effectively integrate the contributions of an active standards defining trade association with open source projects and university research programs.

3.6.3 The Potential Role of Open Source in Innovation

The final area outlined for investigation was, 'Can open source contribute to collaborative innovation between firms in the TMForum?' The issue is best discussed by considering again the organisational relationships, previously outlined in the discussion of Figure 8 in Section 3.6.1, in the context of Eric von Hippel's research into open source as a tool for collaborative innovation (*von Hippel, E. 2005*), (*von Hippel, E. 2001a*).

One of the reasons service providers prefer COTS products over bespoke development is that the cost risk is quantifiable in terms of the licence conditions through the life of the contract. However, service providers entering long term COTS supply contracts also often choose the vendor based on an expectation (and sometimes explicit commitment) to follow a particular product development road map. Therefore although the cost might appear to be controlled, the future delivery of expected functionality is still a risk to both parties. The risk is less than bespoke development if a solution can be specified in terms of simple incremental improvements to the existing COTS technology. However if the requirement goes beyond current product capabilities, the risk profile looks more like that of bespoke development.

If the solution requires changes to more than one COTS product, the development risk is multiplied by the number of products since the projected solution will fail if any one of the products does not deliver to their roadmap expectations. As discussed in Section 3.4.3.2, ISV's sometimes merge to build a portfolio of products which can be made to follow an integrated roadmap. Alternatively the larger Integrators can be contracted to manage this risk on the service providers behalf by corralling a portfolio of 'best in class' COTS products for which they maintain the unified roadmap and vendor relationships.

Although these solutions deal with the contractual risk, they do not deal with risk of incorrectly interpreted requirements. If a COTS vendor is required to commit to future

---

2007]

[37] MTOSI Reference Implementation http://sourceforge.net/projects/mtosi-ri [Accessed Sept 2007]

development of uncertain functionality against a fixed price, they are likely to add a risk contingency into the price. In the end they are not guaranteed to deliver what the service provider really needs because, although they meet the letter of the specification, they may still fail to appreciate the underlying requirements (*Chapman, C. B. et al. 2003 p271 ff*).

(*von Hippel, E. 2001b*) has noted that in general a manufacturer cannot expect to get a solution right first time because users do not usually know exactly what they need at the start. Furthermore, some issues are impossible to foresee prior to use. The details and subtle interactions of product use cannot be fully captured in a specification but these problems will emerge when a new product or service is placed in production. One can thus conjecture that even an NGOSS component compliant with an NGOSS contract specification will have aspects which cannot be fully or accurately specified as was argued in Section 1.4.

(*von Hippel, E. 2001b*) has noted that it is often a very costly matter for firms to understand users' needs deeply and well; user need information is very complex and conventional market research techniques only skim the surface. One practice in the OSS COTS ISV community is to assign product consultants to large service provider accounts with the objectives of helping the customer get the most out of the product, providing user feedback to the product managers and developers and fostering new sales opportunities.

This practise in some senses corresponds to what von Hippel refers to as an 'ethnographic' study of user needs (*von Hippel, E. 2001b*), where the consultant 'lives' alongside the users as a representative of the OSS vendor. Ideally product consultants will have had prior experience of the working environment in which the product is to be used and can relate closely to the needs of the customer. The author has had direct experience of such a role as a product manager seconded for a year into a service provider. This approach has a number of draw backs. It is expensive (usually funded out of cost of sales) and it relies on the ability of the product consultant to effectively champion customer needs back into a development organisation which can be resistant to marketing input for the reasons outlined in Section 3.6.1.

Von Hippel has suggested that the root cause of the problem is that often the information used in technical problem solving is 'sticky information' i.e. costly to acquire, transfer, and use in a new location (*von Hippel, E. 1994*). Despite attempts to standardise operations processes through the eTOM, effective network management remains something of an art. How users apply OSS tools is often tacit knowledge - i.e. knowledge that they carry in their heads and is, therefore, difficult to access (*Polanyi, M. 1967*).

However a process of encouraging users to use prototypes in their problem space can lead to highly innovative feedback as the users learn how to use the technology. (*von Hippel, E. & Tyre, M. J. 1995*) first investigated this innovation process by examining how a new generation of machine tool was introduced to a factory setting. They noted that real life industrial problems were rarely structured in that they often had an unknown solution space (a precisely

specifiable domain in which the solution was known to lie). Such problems were solved by a process of first generating several alternative solutions. The process by which users and manufacturers learnt from using prototypes was more intentional than just 'trial and error'. Rather, users exhibited a form of complex pattern matching (which von Hippel termed 'Templating') in which two very different and highly complex patterns, the new machine and the plant context, were brought into close juxtaposition during field use and the examination of the interference between these objects led to innovative improvements.

(*von Hippel, E. 2001b*) has noted that several industries are introducing user toolkits in order to allow users to produce actual products or services prototypes (most importantly) at the user's site. Von Hippel first examined this in LSI Logic's introduction of software to allow users to design and test their own logic designs before committing to the manufacture of programmable gate arrays. He has observed the same principles in other industries such as food manufacturing. A well designed toolkit transfers only the 'needs related' design to the users, while retaining the complex industrial manufacturing process within the manufacturer.

While many middleware vendors provide trial download components for user evaluation, the idea of a tool kit goes much further in that it requires there to be a direct feed back mechanism for modifying the core commercial product. In some senses OSS/J could be considered an innovation toolkit in the OSS space, since it has allowed users to design and prototype useful interfaces which have subsequently been productised by OSS vendors.

(*von Hippel, E. 2001a*) makes several arguments as to why open source could lead to greater (or more useful) OSS product innovation. Outsourcing OSS development to COTS ISV's, results in service providers paying an 'agency cost' to the ISV's because their incentives don't match those of the customer. This often takes the form of the product not being the best possible fit to a user's particular needs because an ISV wants to make the product as attractive to as many users as possible.

Von Hippel describes difficult user group meetings where vendors urge users to compromise in prioritising common requirements which fit within the vendors' development resource budgets for future product releases. As an OSS Product Manager, the author has had painful first hand experiences of conducting such meetings. This is particularly difficult as a product becomes a 'cash cow' or is approaching the end of its life. The vendor is unprepared to invest heavily in product development - a fact which they would rather not reveal to current or new customers.

The findings of (*Torchiano, M. et al. 2004*) discussed in Section 3.5 tend to suggest that customers would prefer to extend rather than compromise on COTS functionality and open source may provide a route for cost effective extensions. (*Ruffin, M. et al. 2004*) describes open source as potentially helping with end of life issues because customers can continue to

develop product functionality using the open source model after the manufacturer loses interest.

(*von Hippel, E. 2005*) suggests that innovation is actually retarded if manufacturers can only afford to manufacture for a mass market since the innovations pioneered by today's innovating lead users will be adopted by the mass market tomorrow (Figure 6 Section 3.3.1). A more generalised product can be realised if the development process can accommodate a variety of lead users in different contexts. Open source can encourage and allow many innovating lead users to contribute ideas to the product.

Section 3.4.2 noted that R&D in service providers is now focused on helping them learn to use, rather than learn to produce technology. Therefore to create a technology to solve a given problem, they need to actively foster collaboration with their specialist technology suppliers. Open source can assist both sides of this collaboration by inherently providing a vehicle through which to manage the risks of correctly specifying functionality.

However to use open source in this way requires two things to happen. Firstly, the COTS OSS vendors need to embrace an open source development model. Section 3.4.3.3 described how this already happening in the middleware market and it was noted that a lot of useful NGOSS functionality can be delivered using middleware alone.

Secondly the role of the R&D function in service providers needs to change to embrace the user role in an open source development model. This does not imply that the service providers would need to take ownership of the entire development process; rather they would need to strategically engage with the open source projects which acted as the foundations of their chosen ISV's products in order to ensure that their requirements are fully understood and tested by users in a representative trial environment.

If one characterises a more comprehensive TMForum open source program as fulfilling a similar function to an innovation toolkit, a number of requirements emerge for such a program which are listed in Table 25.

Table 25 OpenOSS Requirements Derived From Innovation Toolkit Requirements	
Toolkit Requirements (von Hippel, E. 2001b)	OpenOSS requirements
1. Must help users produce actual prototypes products or services at user site.	• Must provide a capability which can be reproduced close to end users and used in trial service environments.
2. Allow users to try complete cycles of trial and error learning.	• There must be a feedback mechanism to allow user ideas to be incorporated in future developments.
3. Offer users a solution space that encompasses the designs they want to create.	• Must address the OSS aspects of a larger 'playground' for prototyping service creation including service delivery components.
4. Users able to operate them in their customary design language and skills - user friendly - no need for additional training.	• Available to users in the form of easily installed and usable applications with support and documentation. Does not require user to be a developer.
5. Provide libraries of commonly used modules - allow user to focus on truly unique areas of design.	• Uses well designed of-the-shelf component technologies which allows innovation to concentrate on the OSS specific functionality.
6. Will ensure that custom products and services designed by users will be reproducible in manufacturer production equipment without revisions by manufacturer based engineers.	• Uses core technology components which although open source either has a standardised and therefore commercial substitute (e.g. J2EE) or has a direct commercial support route (e.g. OpenNMS , Drools).

3.7 Research Program Design

The preceding sections have laid out the Action Research epistemology in terms of a diagnosis of a real world problem (P) and a Framework of ideas (F). This theorises how the collaborative innovation fostered in standards organisations might be combined with open source and applied as a Problem Solving Methodology (Mps) to reduce the risks of new product introduction - both in Service Providers and in OSS ISV's.

Obviously no single intervention could contribute to the understanding of all of the issues outlined. However the research program has covered much of the problem space using four projects;

OpenOSS Catalyst	Investigates the institutional aspects of using an open source project in the TMForum.
NGOSSoil	Investigates an attempted vendor contribution of an open source NGOSS component through the TMForum.
Arqiva DVB-T PoC	Investigates collaborative and user innovation using open source while architecting a business critical OSS project.
OpenNMS OSS/J	Investigates practical OSS open source development through participation in an active open source OSS project (OpenNMS).

Table 26 summarises the key characteristics of each intervention. The specific contributions of each intervention will be discussed in the remainder of the Thesis.

Table 26 Key characteristics of each project intervention

	Intervention			
Project Aspect	**OpenOSS Catalyst**	**NGOSSoil**	**OpenNMS OSS/J**	**Arqiva DVB-T PoC**
Primary area of investigation	Institutional aspects of an Open Source project in the TMForum	Attempted vendor contribution of an Open Source NGOSS component through the TMForum	Participation in an active Open Source OSS project (OpenNMS)	Collaborative user innovation while architecting a business critical OSS project
Objectives	• Explore the benefits and issues of using Open Source in the TMForum • Launch a TMForum community of interest in Open Source • Explore collaborative innovation using Open Source in a Catalyst project • Model an Open Source solution using NGOSS MDA project	• Attract developers from service providers, Integrators and ISV's • Build business case for starting the project • Plan project implementation	• Learn the realities of Open Source OSS development from participating in a real project • Create an advanced prototype of and OSS/J Qos interface for OpenNMS	• Create a proof of concept (PoC) demonstrator at minimal cost • Gain architectural buy-in from users, management and vendors • Inform a real DVB-T project RFQ and tender process
Project type (Table 17)	Glue-Code COTS Based Application (GCBA)	Launch of Open Source project to create a genuine NGOSS component	Non-COTS intensive (component development)	Glue-Code COTS Based Application (GCBA)
NGOSS project type	**NGOSS Wrapper Technology:** Use of NGOSS derived 'glue ware' to solve the problem of integrating disparate applications by standardising on a common communications environment	**NGOSS Open Source project;** Launch Open Source NGOSS component (NGOSSoil) (potentially useful as wrapper technology)	**NGOSS Component Technology:** OSS/J Qos Interface integrated into OpenNMS	**NGOSS Proof of Concept:** Use NGOSS standards (SID, TAM, eTOM etc.) as guidelines for proof of concept design.
Primary stakeholders	SP's: BT, Vodafone, Colt, NTT, Covad, QinetiQ ISV's: Agilent, Sun, Automagic KB Universities: UoS, BUTE Integrators: Invocom, Cognizant Trade Association: TMForum	SP: BT Universities: UoS Equipment Vendor: Nortel Integrators: Invocom, Cognizant Trade Association: TMForum	ISV: OpenNMS Universities: UoS	SP: Arqiva ISV's: Sidonis, OpenNMS Universities: UoS Integrators: Invocom, Scysis
Service Application layer	VoIP service	Multi-technology	IP/SNMP device management through OSS/J	Digital TV (DVB-T) SLA Management
eTOM processes	Service Assurance	Service Assurance	Service Assurance	Service Assurance
COTS/Open Source mix	Mixed COTS / Open Source	Mixed COTS / Open Source	Open Source	Mixed COTS / Open Source
Granularity of project software; (components Vs applications)	• Large COTS Application: Agilent NgnAS • Medium Open Source Applications; OpenNMS, OTRS, SER, Nistnet • Open source components in glue-ware; JBossAS, Drools, JBoss Portal	• NGOSS component developed originally by Nortel using mixed Open Source / COTS component software	• Medium Open Source Application: OpenNMS. • Open source components in architecture: JBossAS, JBossMS, XMLBeans, Spring, Maven, Hibernate, PostgreSQL, Castor, SNMP4J, JRobin, Tomcat, Fedora Core 4	• COTS application constructed using Open Source components: Sidonis StateWise • Open Source application: OpenNMS • Open Source components in glue-ware; JBossAS, JBossMS, XMLBeans

Table 26 Key characteristics of each project intervention

	Intervention			
Project Aspect	**OpenOSS Catalyst**	**NGOSSoil**	**OpenNMS OSS/J**	**Arqiva DVB-T PoC**
Open Source Development	• Open Source applications used as COTS components and extended using glue-ware (no functionality extension) • COTS application extended using Open Source glue-ware	• Potential code preparation in order to Open Source an application internally developed by Nortel	• Functionality extension and interface development for OpenNMS	• Systems architecture development • Glue-code development • Model development and Configuration • Proof of concept testing and demonstration
Practical use of NGOSS to analyse problem.	Modelling of Open Source solution using NGOSS UML model (Automagic)	Proposed modelling using NGOSS in use cases	OSS/J as an implementation of NGOSS wrapper	NGOSS models used to analyse DVB-T SLA problem
Industrialisation Strategy	• Using mainstream e-business technology standards • Possible to industrialise using commercial -OSS/J J2EE etc • Service platform could be extended to OpenIMS next generation services	• Community & commercial support for imbedding in solutions - Element Managers, Inventory Models and Mediation • Nortel intend to use the software in future development	• Contribute development into OpenNMS code stream	• Inform DVB-TV project specifications • Productisation of Sidonis StateWise and Drools integration
IPR - licensing	• TMForum Catalyst IPR rules • Apache 2 Licence for Open Source	• Apache 2 Licence for Open Source • (issue with imbedded COTS components)	• GPL for OpenNMS extensions • Apache 2 for OSSBeans code	• Community source model for specific proof of concept created under NDA agreement.
Team integration strategy	• Remote team integration using ICT (Scotland, England, France, N America, Budapest, India) • Face to face meetings • Mail exploders • Project Wiki Portal • Sourceforge CVS server (OpenOSS) • VPN and virtual lab • Skype / Bridge for conference calls	• TM Forum Collaboration Work Space • Mail exploders • Face to face meetings • TM Forum gForge server	• OpenNMS Dev Jam & meetings • Sourceforge SVN server (OpenOSS) • Sourceforge SVN server (OpenNMS) • OpenNMS wiki / dev lists • OpenNMS IRC channel • Skype / Bridge for conference calls	• VPN and virtual lab • Team Wiki • Bridge for conference calls • Face to face meetings (OpenNMS)

Chapter 4 Intervention: OpenOSS in the TMForum

Chapter 4 Intervention: OpenOSS in the TMForum

4.1 Introduction

Chapter 3 posed arguments suggesting that open source could aid OSS practitioners strategically manage the profitability of the new product introduction process through increased risk efficiency, greater collaborative innovation and accelerated creation of an NGOSS component market place. This chapter considers two specific Action Research interventions within the TMForum designed to test sub sets of these arguments using open source as the problem solving methodology (Mps) under examination.

OpenOSS Catalyst	Investigates the institutional aspects of creating an open source project in the TMForum
NGOSSoil	Investigates an attempted vendor contribution of an open source NGOSS component through the TMForum

The OpenOSS catalyst and NGOSSoil projects took a year and a half to plan and execute. The history of the initiation, planning and execution of the projects is covered in Appendix A.1 and will not be repeated here. Instead this chapter will discuss the Action Research design and results of these projects in terms of the wider industry problem space outlined in Chapter 3.

4.2 The OpenOSS Catalyst Project

The OpenOSS catalyst intervention was designed to create a simulated project environment which, as far as possible within the limitations of a TMForum Catalyst, was representative of a commercial telecoms OSS project that had chosen to use Open Source as a key aspect of its development approach. It was hoped to create a scenario which would shed light on how realistic OSS projects might be modified by the introduction of viable open source components complementing the COTS components in the OSS market place.

4.2.1 Diagnosis

Given the OSS industry's prior indifference to Open Source, the researcher's ability to establish a credible Open Source Action Research project incorporating leading OSS industry players represented a significant challenge. Section 3.3 outlined an approach to managing the risks of establishing a successful Open Source project by ensuring that the proposed project design reflected the aspirations of each of its stakeholders.

The Diagnosis phase of the OpenOSS project was concerned with understanding the drivers motivating the potential players so that the Action Planning phase would create a win-win situation in which all of the stakeholders' expectations were met.

The analysis in Chapter 3 which looked in general terms at the whole OSS Component Market Place was in part informed by the Diagnosis phase of this project which was conducted in the context of the negotiations to get the stakeholders involved. However along the way, several parties showing initial interest decided to drop out. The Diagnosis reported here is not focused

the general market but on the actual OpenOSS participants. It represents their specific motivations for committing to and continuing with this particular project.

The OpenOSS project stakeholders were major industry players drawn from the mainstream OSS Component Marketplace who already had a credible history of commercial engagements with each other. Although the sample of the OSS industry was 'self selected' by the participants' individual decisions to join the Catalyst, they were not atypical of other OSS industry players. The individual stakeholder motivations and contributions which moulded the design of the project are summarised in Table 27. A detailed analysis of the motivations of the stakeholders is provided in Appendix D which also describes BT's Next Generation network on which the Catalyst was based.

Table 27 Open OSS Project Analysis Using 6 W's Analysis framework
(Chapman, C. B. & Ward, S. 2003 page 4)

Who (parties)		Why (motives) What do the parties want to achieve?	What (design) What is it that the parties are interested in?	Which way (activities) How is it to be done?	Wherewithal (resources) What resources are required?	When (timetable) When does it have to be done?
Service Providers	BT	OSS cost reduction Less Integration Tax NGOSS realisation	Encourage ISV's and other SP's to investigate open source	Actively participate in developing OpenOSS catalyst project	SP Sponsor of Catalyst 1 Developer to guide students 1 Senior investigator to lead project Student sponsorship funding	Open source is tactical move in strategic NGOSS standards program Catalyst: Jan - May 2005
	Vodafone	Wider OSS/J adoption Pre-tender verification of NGOSS standards	Promote university research into practical problems of using OSS/J	Participate in Catalyst follow up meetings	SP Sponsor of Catalyst Part Fund university participation	Open source is tactical move in strategic OSS/J program Financial support to UoS
	Covad	B2B integration cost reduction through community developed OSS/J interfaces	Build community for maintaining OSS/J interface adaptors	Influence catalyst project content	SP Sponsor of Catalyst	Catalyst: Jan - May 2005
	QinetiQ	Low cost NGOSS demonstrators for promoting NGOSS to MoD customers	Encourage the development of open source components for use in MoD demonstrators	Influence catalyst project content	SP Sponsor of Catalyst Link NGOSS MDA catalyst to open source catalyst	Catalyst is a tactical project in TMForum marketing plan Catalyst: Jan - May 2005
	C&W	Low cost OSS options for foreign properties (Caribbean)	Investigate open source solutions	Influence catalyst project content	SP Sponsor of Catalyst	Catalyst: Jan - May 2005
	Colt	Reduce cost of B2B e-business with European PTTs	Investigate open source as an enabler for B2B integration	Influence catalyst project content	SP Sponsor of Catalyst	Catalyst: Jan - May 2005
	NTT	Share experiences with OSS open source community	Complement NTT's current use of open source in Japan	Influence catalyst project content	SP Sponsor of Catalyst	Catalyst: Jan - May 2005
ISV's	Agilent	Promote new NgnAS product Get SP feedback on prototypes Leverage OSS commoditisation	Actively engage as ISV in open source program in TMForum	Lend COTS product (NgnAS) Contribute glue code to open source demonstrator Assist students with J2EE Lead marketing effort	1 Marketing manger 1 Architect Student sponsorship funding	Catalyst is a tactical project as part of Agilent's university liaison program. Catalyst: Jan - May 2005
	Sun	Promote OSS Middleware and OSS/J adoption	Find ways to encourage service provider adoption of J2EE middleware and OSS/J	Encourage open source OSS projects using JCP standards; OSS/J, J2EE, Rules engines	Help with OSS/J and marketing	Use Catalyst as a tactical project as part of OSS/J marketing program
	Automagic	Proof of concept for NGOSS UML tooling	Demonstrate NGOSS lifecycle view of project implemented using open source	Take the developed OpenOSS project and reverse engineer an NGOSS model for it.	Modelling of Open Source project	Catalyst a tactical project for Automagic consulting company Catalyst: Jan - May 2005

Table 27 Open OSS Project Analysis Using 6 W's Analysis framework
(Chapman, C. B. & Ward, S. 2003 page 4)

Who (parties)		Why (motives) What do the parties want to achieve?	What (design) What is it that the parties are interested in?	Which way (activities) How is it to be done?	Wherewithal (resources) What resources are required?	When (timetable) When does it have to be done?
Integrators	Invocom	Build Integrator / Consultancy business	Build Invocom brand awareness Demonstrate thought leadership through EngD sponsorship	Be seen as founding partners in developing the open source OSS project Provide project management expertise	1 Project manager 1 Networking expert Loan of routing equipment Student sponsorship funding	Catalyst a tactical project within 4 year EngD research sponsorship. Catalyst: Jan - May 2005
	Cognizant	Build Integrator / Consultancy business	Opportunity to showcase Indian Open Source practice	Be seen as active contributors to an open source OSS project Provide integration expertise	1 Catalyst integration manager 3 Indian developers	Catalyst is a tactical project in TMForum marketing plan Catalyst: Jan - May 2005
Universities	UoS	Research opportunities Student projects Funding	Build relationship with Telecoms industry through the TMForum	Eng D Action Research Contribute undergrad projects Bid for collaborative funding	1 Research Engineer 3 student projects Lab facilities Wiki, VPN 7 servers	Catalyst is a tactical project in 4 year EngD program Catalyst: Jan - May 2005
	BUTE	Research opportunities Student projects Funding	Build relationship with Telecoms industry through the TMForum	Contribute undergrad projects Bid for collaborative funding	2 student projects 1 lecturer	Catalyst: Jan - May 2005 Build on existing research group funded by PTT in Budapest
Trade Assn.	TMForum	Demonstrate value to members through innovative projects and realisable standards	Demonstrate practicality of NGOSS Enhance Catalyst program Explore open source Keep ISV, SI and SP members happy	Encourage catalyst project to explore open source NGOSS Agree open source strategy at board level	Waive catalyst participation fee for Universities Assign 1 staff member to open source program	Catalyst: Jan - May 2005 Leverage Catalyst to establish a recognised TMForum open source program

4.2.2 Action Planning

Having used the Diagnosis phase of the project to understand the motivations of the potential stakeholders and define the problem space, intensive Action Planning took place between the OpenOSS planning workshop (2/12/2004) until the end of the Team Action Week in Portugal (02/02/2005).

From a research perspective, the aim of the Action Planning phase was to create a project scenario which would allow the team to investigate the institutional aspects of initiating and running an open source project within the TMForum. To do this it was necessary to simulate as realistically as possible a scenario for the creation of an OSS open source project, albeit within the time and resource constraints of a TM Forum Catalyst project.

The Action Planning phase of the OpenOSS project sought to create a win-win situation in which a project was designed which would be supported by enough stakeholders to make the project viable and at the same time generate useful results. By the time of the planning workshop, company champions had been identified who (by their attendance) had indicated a willingness to at least try and get their companies involved. However it was necessary to create a scenario which provided enough tangible business benefits for each of the champions to be able to convince their management to offer support.

The following sections will describe how the Catalyst was designed to meet the stakeholder expectations. For more details about the technical content of the Catalyst see Appendix G.

4.2.2.1 Expected Risk Profile of the OpenOSS Catalyst

The OpenOSS Catalyst project was designed to deliberately 'stress test' how a team drawn from the OSS industry would respond to using open source and whether it allowed Universities to overcome the natural barriers to academic involvement in the TMForum innovation process.

In Section 3.3 a framework for analysis was proposed to examine the impact which a Development Approach combining COTS and Open Source from OSS Component Marketplace would have on an OSS project's risk profile.

The OpenOSS Catalyst was designed to have an exceptionally low 'expected cost' of innovation through the use of Open Source Components. With the exception of the single COTS component, Agilent's NgnAS, the entire demonstration could be re-constructed by anyone with access to the internet. The cost was also minimised through the use of students as developers and through the use of developers based in India (although the actual cost of these resources was not exposed by Cognizant). Unusually for a Catalyst, one service provider (BT) helped in reducing the risk by committing a development resource to assist the students.

Although the 'expected cost' of innovation was lower, the risk profile of the OpenOSS Catalyst was worse than other types of OSS project as illustrated in Figure 10 and described in Table 28. Even though having one mature COTS component supported by Agilent's OSS

Architect helped, the inherent risk in the project was much higher than a typical Catalyst because of the inexperience of some of the developers and the lack of familiarity with the open source components used.

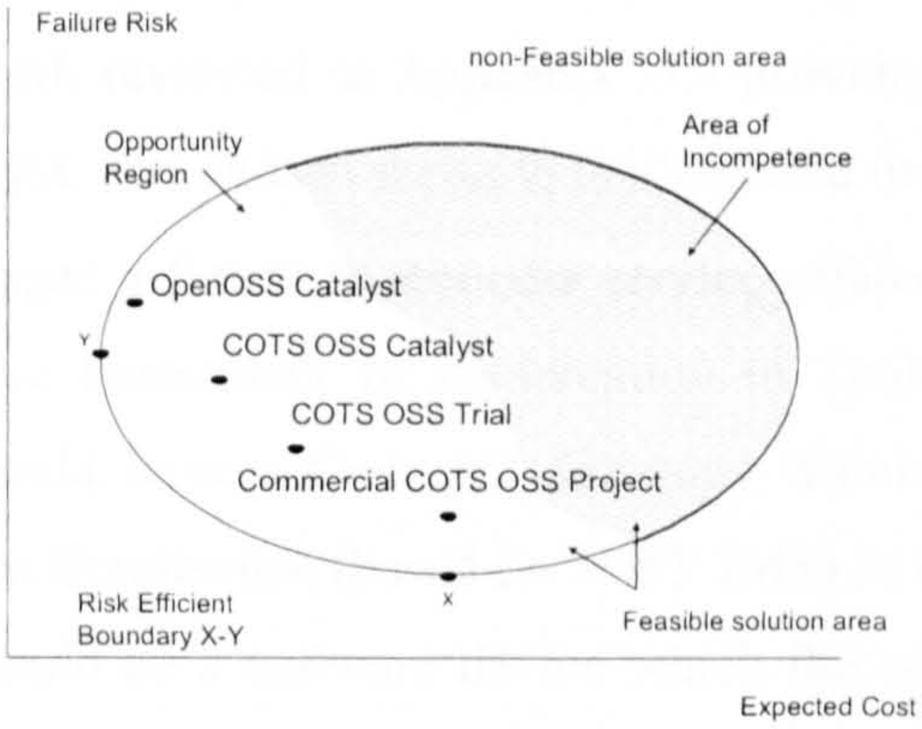


Figure 10 Risk Profile of OpenOSS
(after Chapman, C. B. et al. 2004)

Table 28 Risk characterisation of typical OSS projects
Commercial COTS OSS project • Risk - Low - vendor manages risk • Cost - High - vendor sets price for carrying risk • Notes: A commercial COTS OSS project is typically expensive but the service provider recognises they are paying a premium to minimise the risk of project failure. As BT note in (*Bruce, G. L. et al. 2005a*), the licence cost of COTS OSS products can make niche projects unviable.
COTS OSS Trial • Risk - Medium - shared by Vendors and Service Provider • Cost - Medium - shared by Vendors and Service Provider • Notes: Typically service providers conduct internal trials or proofs of concept prior to committing to a commercial service launch. A trial will include representative service delivery components (the network infrastructure) and the OSS components used to manage the service. The goal is to evaluate whether the functionality of the proposed architecture will perform as required if scaled to a full service. The risk of trial failure is reduced because the COTS vendor supporting the trial knows their product well and supports the service provider in implementation. However, a trail will typically not have to meet the same risk criteria as a commercial rollout. Trials are costly to all parties. Although COTS OSS vendors may support this activity out of cost of sales, the projects are taken seriously and resourced appropriately by both sides. The COTS vendor provides free licences and supports this activity as a necessary step in securing a future sale. If there was no sales prospect, the trial would not happen.
TMForum COTS OSS Catalyst • Risk - High - project can be allowed to fail if business priorities change • Cost - Medium - vendors carry majority of costs • Notes: We can reasonably characterise a TMForum COTS OSS Catalyst as a trial or proof of concept project conducted in the public domain. The collaborative innovation afforded by the Catalyst is used to prove emerging TMForum standards. Often Catalyst participants are also members of TMForum standards generation teams (e.g. MTOSI) and the Catalyst is used to prove the standard by incrementally developing the COTS products. Although development is involved, the risk is minimised through using developers who know the COTS products well and can quickly adapt them to the solution. Although they act as sponsors of the requirements (a minimal burden), service providers do not typically pay towards Catalysts which are funded by the COTS vendors out of marketing budgets (i.e. not by the cost of sales of an identified sales opportunity as in a trial). Catalysts are more risky than private trials since vendors can pull out with minimal loss of face[38].

[38] Two out of the seven planned catalysts in May 2007 Telemanagement World were cancelled because the teams fell apart due to vendor re-prioritisation. Most TMForum activities suffer the same risk of participants loosing interest. Source: Interview with Director of Catalyst program May 2007 (on tape).

4.2.2.2 Open Source Service Network

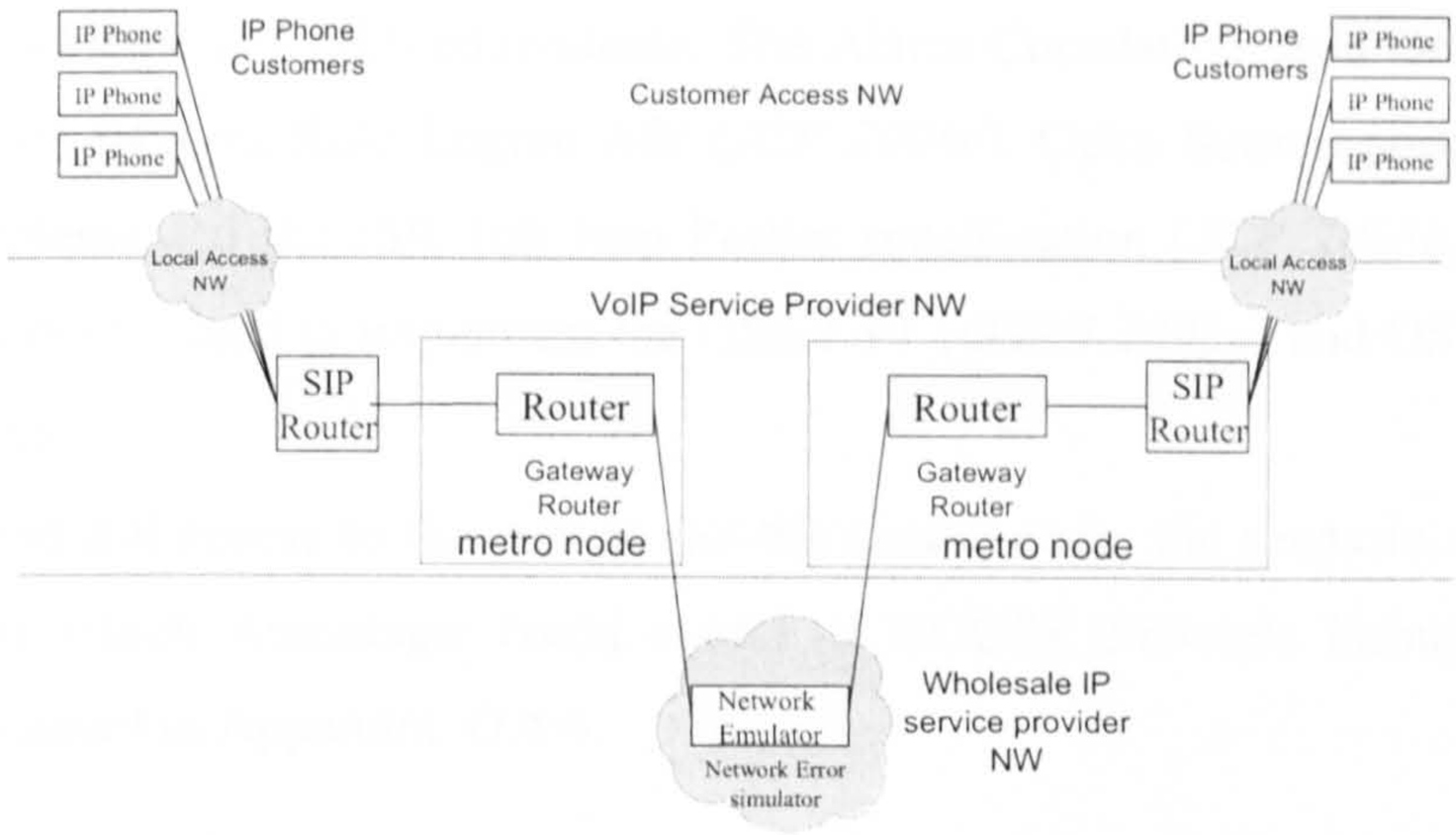
A key objective of the Catalyst project was to create a service simulation using open source components which approximated to a realistic service scenario that might be supported by a next generation network. BT's description of their 21st Century Network project (21CN) and Service Execution Framework reviewed in Appendix D.4 provides a realistic setting against which to compare the Catalyst. The chosen scenario is illustrated in Figure 11.

The service scenario envisaged a future multimedia service offered to the consumer as IP at the service level (i.e. above Fransman's IP demarcation in Table 13 discussed in Section 3.4.2.3). The consumer would have a Customer Gateway which interacted with a service provider's Service Execution Framework (based on 3GPP IMS) in order to access the service. The Consumer Gateway would be a software device which the consumer could obtain from multiple suppliers but which used standard protocols to connect to the Service Execution Framework.

The multimedia service was limited in the simulation to VoIP. An open source SIP phone was used to simulate the Consumer Gateway and several open source SIP servers (soft switches) were used to simulate the Service Execution Framework described in Section D.4.1. This provided a scenario somewhat like that in Figure 31. Although the simulation was only offering VoIP, one could characterise the simulation as representing a cut down version of the Service Execution Framework in Figure 32 because it used the Fraunhofer Focus SIP Express Router (SER) (*IPTEL 2007b*) and SIP Express Media Server (SEMS) (*IPTEL 2007a*) which are at the core of Focus's open source project to create a 3GPP IMS solution, OpenIMS (*FOKUS 2007*), (*Magedanz, T. et al. 2005*), With more resources and time the Catalyst could realistically have been extended to look at IMS using open source.

Using BT's terminology, the Catalyst envisaged a national service provider who provided trunk IP connectivity using an MPLS core but delivered new services as IP at the service level. An open source network simulator (*Nistnet 2007*) was used to simulate the IP connectivity provided by the MPLS core. This could introduce quality perturbations such as lost or delayed packets in the core.

If BT were offering the service, the Service Execution Framework would be hosted at the Metro Nodes. If an alternative operator was offering the service they would interconnect with BT's core network using IP at the MetroNodes but have their own Service Execution Framework which interacted with BT's. This scenario recognised the opening of the market to allow service providers to offer services using complementary services from other providers as discussed in Section 3.4.1.



*Note – the scenario is simplified by not including management of the the Local Access NW

Figure 11 VoIP Network Scenario

4.2.2.3 Open Source OSS Stack

The OSS stack used in the catalyst is illustrated in Figure 12. The stack was envisaged as a simple solution for a niche alternative operator who used the services of a core service provider and needed to monitor the IP streams in and out of their network and also the correct operation of their SIP servers.

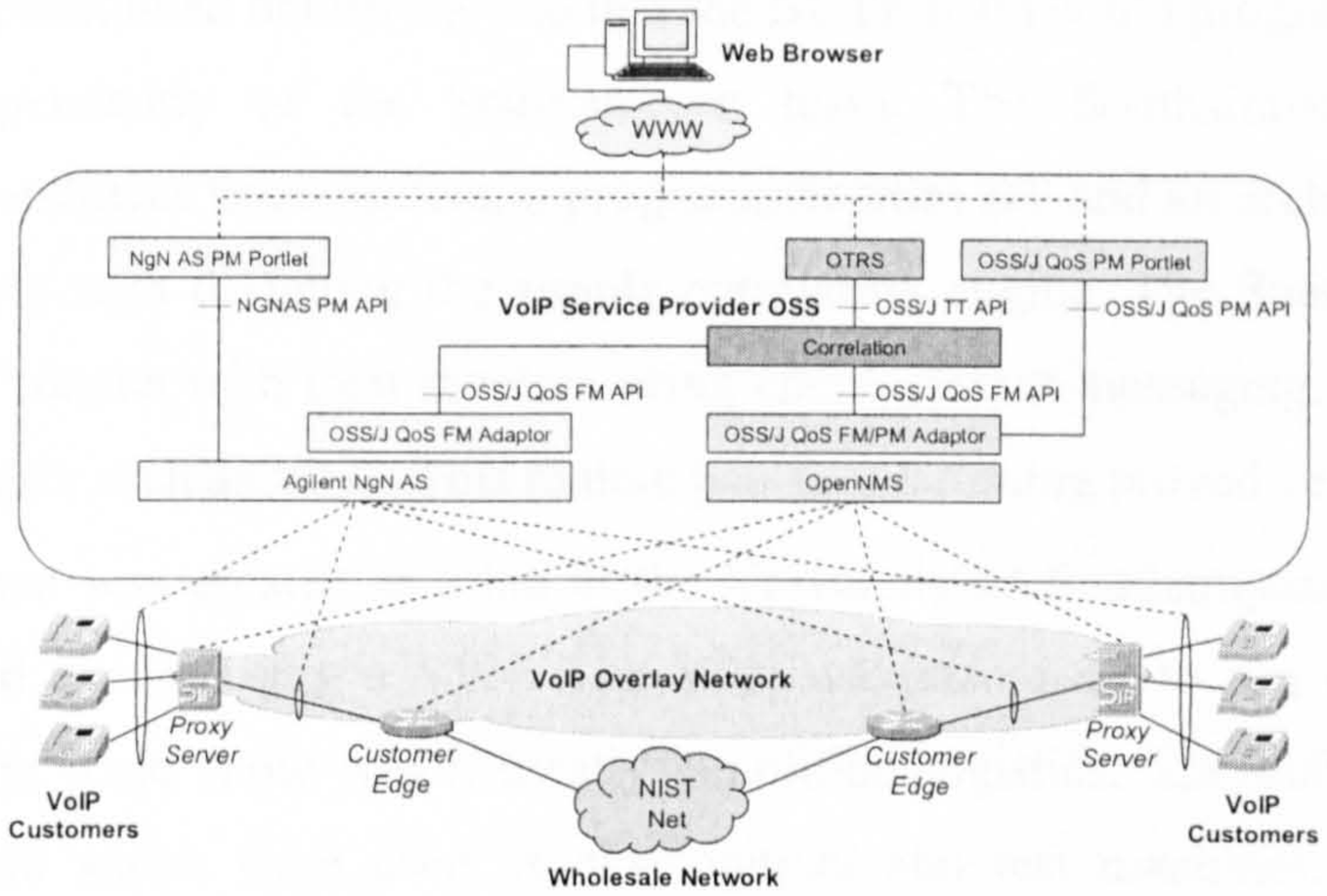


Figure 12 Components in OpenOSS catalyst

The scenario showcased a simple open source OSS stack using OpenNMS, OTRS and JBoss. It met Agilent's requirement to showcase their product and it also met BT's requirement to demonstrate how open source interface implementations could compliment a COTS product and speed up the process of integration. JBoss, an open source J2EE application server was used to implement the OSS/J interfaces and represented a similar function to BT's integration hubs discussed in Appendix D.4.3.

The scenario satisfied SUN's desire to demonstrate the use of open source implementations of OSS standards which had COTS equivalents. The Alarm Correlation used Drools which was based on the JSR 94 Java Rule Engine API (*JCP 2004c*). Open Source portal servers were used which implemented the JSR 168 Java Portlet specification (*JCP 2003b*). Open Source J2EE middleware was used to implement the OSS/J TT (*OSS/J 2002a*) and OSS/J Qos (*OSS/J 2002c*) interfaces.

Because they had full access to the design and the source code, the scenario also provided a simple use case which Automagic could model as NGOSS contracts through the NGOSS lifecycle as discussed in Appendix D.4.4.

4.2.3 Action Taking

The main Action Taking phase was performed between the end of the Team Action Week in Portugal (02/02/2005) and the end of the Open OSS Catalyst presentations at Telemanagement World Nice (20/05/2005). The technical details of the project are given in Appendix G.

A CVS repository was created on Sourceforge to allow the participants to share code and a Wiki was used to share documentation and project plans. The developers were located at several sites;

- University of Southampton, Hampshire, England
- BT Marthlesham, Essex, England
- Agilent, South Queensferry, Scotland
- Cognizant, India
- BUTE Budapest, Hungary

The project was partitioned deliberately so that the BUTE team could progress their work with OpenNMS independently of the Southampton team. The Southampton students were supported by an architect from Agilent, a programmer from BT and an architect from Sidonis who provided help with designing the simple correlation engine. The Southampton students were in constant contact with their mentors using email, instant messaging, Skype and screen sharing technologies such as VNC. This remote pair programming proved very effective.

A test environment was created in a lab at the University of Southampton which all of the participants could access using a VPN. The VPN was also used to run the demonstration remotely from the trade show which greatly simplified logistics. The university supplied a number of servers which were used as development and test machines. The VPN router network was designed by Invocom and lent to the University. Agilent shipped a SUN server to Southampton pre-loaded with their NgnAS software on which they did their project specific development remotely using the VPN. Accessing Sourceforge and the VPN proved difficult for several of the commercial partners since the corporate security policy on their PC's prevented installation of VPN clients or their firewalls did not allow permit access to external VPN's. For this reasons, some commercial participants were forced to work at home.

A project manager was appointed from Invocom who constructed a project plan, maintained a risk register and ran weekly project meetings in a manner similar to any commercial OSS

project. The meetings were conducted using a telephone bridge or Skype. The natural inclination of the commercial participants was to create more project documentation than was necessary for a project of this nature. The initial project definition document and the written integration plan proved to be too detailed and quickly became out of alignment with the reality of the project and were effectively ignored by the developers. However the project manager's regular calls to check progress and his logistical liaison with the organisers of the Catalyst Showcase at Telemanagement World proved invaluable.

The project was time constrained with the deadline to deliver something for Telemanagement World. Weekly technical meetings reviewed progress and design issues with the developers. Functionality was dropped or changed when it became apparent it was impossible to deliver within the time and resource constraints. Initially the Catalyst wanted to deliver a more sophisticated trouble ticketing function, but the student working on this dropped out of his BSc course for personal reasons and could not be replaced. The project also had several abortive attempts to use different portal servers and application servers before settling on JBoss.

Initially it was planned to have the team in India take the contributions and create an integrated project, possibly as a CD which could be offered as a demo at the show. Unfortunately the links between the Indian team and the teams at Budapest and Southampton were never very effective and the final integration ultimately ended up being done in Southampton with the help of one of the Indian team. Clearly the Indian colleagues were working very hard to try and pull together an integration plan using their own processes. However they tended not to proactively approach individuals in the rest of the development team and ask how they could help. While they appeared to be evaluating snapshots of the code on Sourceforge, they did not always provide effective feedback.

The issues exposed when working remotely with the Indian team appeared to derive from an overall lack of clarity about the project structure, rapid churn in the project design and the Indian developers' cultural inclination not to communicate with developers outside their organisation except through formal channels. The time zone differences also had an impact. Similar issues occurred with the Budapest team but had less of an impact because they had a self contained and well defined activity which they could progress without reference to the other developers. The obvious lessons one can draw from this experience are that; a) The fact that the code is shared does not relax the need for clarity in project management and the need to pay attention to the cultural differences across an international project; b) Some of the problems of lack of central coordination can be overcome by clearly partitioning the design.

The primary purpose of the OpenOSS catalyst was to move the open source agenda forwards within the TMForum. Although the technical lessons from the exercise were useful, a communications plan had to be agreed which would effectively leverage the Catalyst in order to prime the next steps in promoting an open source program. Because the marketing content was a significant aspect of the catalyst, weekly marketing meetings were held to agree

common marketing messages and review the progress members were making on the catalyst presentations and position papers which would be given at the trade show. This activity engaged as many resources as the development activity. The importance of the marketing activity obviously derived from the prominence of the Catalyst at an important trade show and the desire of the large companies involved to signal correctly their motivations for getting involved.

The week before the show, all of the show participants assembled in Southampton to complete the integration and rehearse the presentations. The Catalyst cannot be cited as an exercise in creating reliable code. Last minute modifications and bug fixing were being performed right up until the day the show started. The working demo illustrated the concepts and supported the marketing messages but was somewhat unreliable. A student was employed in Southampton to be available to reset servers if anything failed during the show. (Subsequent student work during the summer of 2005 increased the reliability and replicated the demo in BT's premises).

The team gave 5 formal Catalyst program presentations and demonstrations per day over the three days of the show. In addition they gave many private presentations to interested individuals and gave presentations on the OSS/J stand and TMForum booth. Half way through the show an evening meeting was held which presented the work to senior directors from BT, Agilent, TMForum, Vodafone, COLT, C&W and HP. This was with a view to priming discussion on the next stage the project should take.

A final wrap up session on the last day of the show was used to get participants feedback which was contributed to the technical report prepared for the TMForum (see Appendix G).

4.2.4 Evaluating OpenOSS

4.2.4.1 Immediate Evaluation

A period of evaluation followed the Catalyst at Telemanagement World Nice (20/05/2005). From a tactical marketing perspective, the project had been a success. The Catalyst presentations were well received and all of the participating companies had benefited from the publicity. However the frenetic work leading up to the Catalyst had exhausted the team and it was necessary for each of the members to take a step back and evaluate their commitment through the summer of 2005.

Several members published papers about the use of open source in telecoms derived from the project (*Bruce, G. L. & Mottishaw, P. 2005*), (*Bruce, G. L. et al. 2005a*), (*Gallen, C. R. et al. 2005b*), (*Gallen, C. R. et al. 2005c*), (*Gallen, C. R. et al. 2006*) and the TMForum Catalyst report was prepared (Appendix G). The students who had worked on the project were employed by BT and Agilent through the summer of 2005 to ruggedise the demo and do some additional work which resulted in a couple more publications (*Gallen, C. R. et al. 2005a*), (*Bruce, G. L., Robson, P., & Spaven, R. 2006*).

As an intervention, the project did establish an Open Source agenda in the TM Forum. An open source working group was chartered by the TMForum board under the chairmanship of BT and over 50 contacts met through the Catalyst joined the mailing list. The TMForum created its own source code repository and guidelines for community code development were published which recommended the use of the Apache 2 licence. Unfortunately the TMForum's Board did not fully embrace the idea that participants outside the TMForum could access the repository and recommended that any member contributed code should go through a member only approval process before release. This was a less enthusiastic embracing of open source than the DMTF's WBEMsource initiative (*DMTF 2004*) and although it was difficult to see how this would work in practice for developers, some steps forward had been made.

The OpenOSS catalyst provided a very good case study in rapid OSS integration using OSS/J. Automagic went on to use the OpenOSS catalyst as a key part of their TMForum training material on the advanced use of NGOSS.

From a standards perspective, the OpenOSS catalyst did highlight the industry perception that NGOSS was a paper standard and had no recognised reference implementations. The Board's recognition that the TMForum needed to move forwards the implementation agenda led to the significant step of the formal merging of OSS/J with the TMForum[39] and the establishment of the Prosspero program (*TMForum 2007g*) which also supported the MTOSI interface. At the same time, OSS/J established an open source site for members contributions on java.net (*OSS/J 2007c*) although this was primarily concerned with creating reference implementations of OSS/J interfaces and not with open source applications.

The Catalyst project demonstrated the viability of a creating a 'Living Lab' where TMForum members and Academia could work on common projects. In particular it showed how the Catalyst program could be supported with a more permanent research infrastructure including network and OSS components.

The OpenOSS catalyst had taught the team a great deal about the difficulty of running an (albeit short term) open source project within the TMForum and had prepared the ground for a more substantial commitment. However through out the summer of 2005 it became clear that the existing sponsoring companies and the TMForum would not be able to put any more resources into a project unless a strong business case could be made. In addition they were not prepared to support the Universities in seeking EU research funding and so it was becoming increasingly difficult to see how the open source agenda could be taken forwards within the TMForum without more substantial resources.

However an opportunity for developing a more substantial open source project within the TMForum began to emerge late in the summer of 2005. This is the subject of the NGOSSoil intervention discussed in Section 4.3.

4.2.4.2 Later Developments

The discussion in Section 3.3 of the benefits of open source as a tool for a risk efficient introduction of new services appears to be correct. The OpenOSS Catalyst proved that it was entirely possible for a small team to relatively quickly create a service stack using open source components. This was further confirmed by industry developments in 2006.

Table 29 Open source components used by Truphone	
OpenSER: Open Source SIP server Asterisk: Open source SIP PBX FreeSWITCH: Open source software telephony application Apache Tomcat Servlet Container MySQL Databases Debian Linux Build	Spring, Hibernate, JSF, Shale - Clay, Acegi for the web site build Roller for blogging Nagios and Monit for status monitoring OTRS for trouble-ticketing Jira for bug tracking

In September 2006 a 'software only network operator' called Software Cellular Network Ltd launched its Truphone service in the UK. Truphone could offer VoIP services over Wifi and, by the installation of some free user installable client software on top of the standard Symbian operating system in Nokia N95 mobile phones, could bypass the normal voice tariffs of Mobile operators. It is informative to note that Truphone's call server stack is entirely open source (Table 29) and an extension to that used in the OpenOSS catalyst (*Truphone 2007*). They have reported using SER and OTRS and are in the process of installing OpenNMS as part of the system[40].

Vodafone, Orange and T-mobile almost immediately reacted by hacking the Symbian operating system delivered with their branded phones and blocking calls to Truphone numbers (*Wikipedia 2007*). This is the subject of an on going legal wrangle as to the right of operators to deny access to a competitor and the right of third parties to install software clients using published Symbian API's.

The fact that a small start-up company using open source software could cause such panic in the industry is very telling. Appendix D.4.1, looks in some detail at BT's Network and Service Strategy, which (in common with other network operators) involves offering all services as IP at the point of access and fostering a market in applications running on a next generation Service Execution Framework based on IMS. However, as Truphone have illustrated, innovative next generation services can already be offered in the mobile IP layer using open source components which bypass an operators voice calling functions.

In the future third parties could bypass or supersede BT's complex SEF (which has yet to be deployed) if it is not adding value and is not easy to use. If it is to be adopted as a network API by future developers, it will have to be price competitive with third party alternatives and easy to program and test. (As discussed in Section 1.3.1, the telecoms industry has an unfortunate

---

[39] The Influence of this work on the development of Prosspero was independently confirmed by interviews recorded at Telemanagement World May 20-24 2007 with Ian Best, Head of TMForum Product Management and Gary Bruce of BT.
[40] Private conversation in 2006 with OpenNMS developer Jonathan Sartin who now works for Truphone.

history of backing complex standards which are bypassed by simpler ones because implementations are freely available - e.g. TCP/IP vs. OSI)

Section 3.4.2 noted the market penetration of open source above the IP layer. By moving access to all services above the IP layer, the incumbent operators have created fast moving competition for themselves. As Google have found, having a coherent open source strategy may be a key factor in continuing to generate revenues from new IP based services.

4.3 Intervention: NGOSSoil Case Study

4.3.1 Diagnosis

Having begun an investigation into the value of open source as a medium for developing OSS solutions, it was now necessary to develop a more substantial project in which to explore the theme further.

The summer student work in 2005 kept the project alive while several of possibilities were investigated. A number of the participants on the OpenOSS catalyst met several times through the summer of 2005 to brainstorm the possibilities for a project. The OpenOSS project had identified champions within a number of organisations such as Doug Carson in Agilent, Jorg Frankinberger in Vodafone and Gary Bruce in BT. However, although very willing to sponsor a project into their organisations, they needed a compelling business case to secure any resources.

From a research perspective, the inability to find a suitable project within the TMForum represented a risk to this EngD research program and so while these discussions continued, alternative open source OSS action research opportunities were also pursued outside of the TMForum such as those leading to the work with Arqiva and OpenNMS.

However an opportunity for developing a more substantial open source project within the TMForum began to emerge late in the summer of 2005 when Nortel approached the team with a proposal to release a substantial body of code. The circumstances surrounding this opportunity will be discussed in the next section.

4.3.2 Action Planning

The detailed history of this engagement with Nortel is given in Appendix A.1. The researcher was given early access to Nortel information under an NDA and assisted in brokering a situation where Nortel felt able to approach the wider TMForum with their offer. The information in this section has subsequently been shared by Nortel with the wider OpenOSS community.

Nortel were in the advanced stages of developing a java based management technology which they called the Multi Service Platform (MSP). Nortel were careful not to position the MSP as an OSS since they did not want to be perceived as competing in the OSS business. Rather, they described the MSP as an Operations, Administration and Maintenance (OAM) technology

capable of managing a sub network; i.e. it could manage a region containing Nortel equipment but was not intended to scale to a whole network including equipment from multiple vendors which would be the domain of an OSS. This positioned it as having more capability than a simple element manager which managed elements as individual nodes. Having an internal network model, the MSP could offer FCAPS management of the relationships between the nodes and services in a region. Although it was not positioned as an OSS, for a small operator, the MSP functionality would meet many of the needs of an OSS system.

At the time when initial discussions with Nortel commenced, they had already invested a significant number of man years in creating the MSP but it would take some time to ramp its commercial deployment by their lines of business. They no longer wished to sustain the previous level of research investment in the project but they wanted to find a mechanism to continue to keep the project fresh and innovative for a year or so while they developed plans to exploit it commercially. Their feeling was that open sourcing some aspects of the project would offer them more options going forwards and would potentially benefit their customers.

The question was, which parts of the platform should be made open source and how could it be done effectively? Nokia, who had attended the OpenOSS workshop in Southampton were in the advanced stages of launching OpenEMS, an Element Management toolkit with commercial support (*Nokia 2007*). Nortel did not wish to provide a similar offering with the MSP. However one of the major costs which Nortel faced as an equipment vendor was the cost of integration into many different customers' OSS systems. Nortel believed that this cost could be reduced for the whole industry if some aspects of the MSP technology were generally adopted as a means to map between different equipment vendors data models and those in the OSS systems.

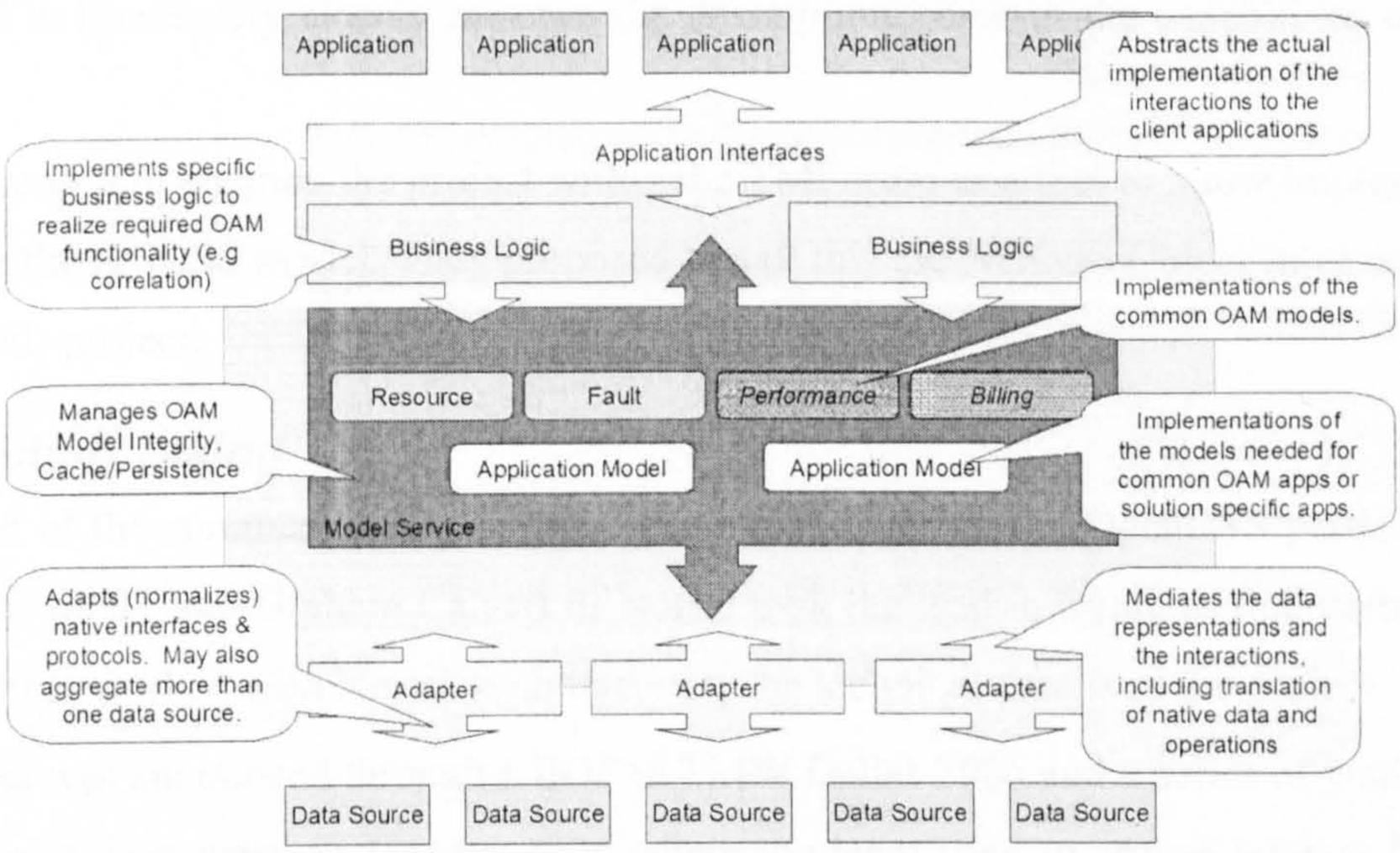


Figure 13 Nortel DML Functional Diagram

(Source: Nortel presentations TMForum Team Action Week 29th Jan 2006

Although the MSP contained many of the components needed to make an element management platform (GUI etc), Nortel proposed to open source only part of the MSP platform, the core model repository or Data Management Layer (DML), illustrated in Figure 13. This was designed to be a repository for any network management model which could be based upon the NGOSS SID or the DMTF CIM. The Meta Model used in the repository could be loaded using XML which could be extracted from a UML model using XSLT transformations on the UML XMI.

Protocol adaptors would sit below the model and interact with the managed devices using protocols such as MTOSI, CLI or even SNMP. In the MSP, Applications such as Network Management Browsers or Alarm Correlation would sit above the repository and interact with the model. However the DML could also support interfaces such as OSS/J or MTOSI above the repository for connection to other management systems. Thus a key potential application for the DML was as a core model repository for a network management system with the ability to map between management models and simplify the problems of integration.

That this capability could be useful to a service provider is illustrated by the discussion of BT's transform program in (*Azmoodeh, M., Georgalas, N., & Fischer, S.*). Although the context of the paper is in justifying BT's interest in MDA, it provides some detailed descriptions of BT's approach to integrating OSS's supporting their new ICT services. BT evidently have put a lot of in house development effort into creating what they describe as a 'data services layer' which provides a virtual repository for OSS applications, where the schema of the repository is the information model (loaded using XML). OSS applications are presented with a common view of data structured according to the information model that sits on top of a collection of management data sources within the OSS. The description of this capability which BT developed in house very closely matches the description given to the capabilities of Nortel's DML.

Nortel proposed to position the project within the TMForum as a tool to allow implementers to instantiate the NGOSS model. They proposed to call this the NGOSS Object Integration Layer (NGOSSoil) project.

4.3.3 Action Taking

At the end of the summer of 2005, Nortel were introduced to the OpenOSS participants and Gary Bruce of BT and Graham Calvert of Nortel took the lead in trying to find partners while the researcher worked with Nortel to characterise the technical details of the project.

The project was announced through a BOF in TMW Dallas 2005 and a series of workshops set up at Team Action week in January 2006 where the team tried to arouse interest from other TMForum members.

Nortel prepared a project plan for releasing the code which required a team of 3 from Nortel to be committed to fostering the development of a community. Some work was required to create

a repository, provide sanitised documentation and ensure that the project could be launched cleanly.

The DML had been designed using some embedded COTS components which meant that the present project could not be considered fully open source. Most importantly SolarMetric's KodoJDO (*BEA 2007*) implementation of the JSR12 Java Data objects API (*JCP 2003a*) which was used by the DML for Object Relational Mapping. One of the exercises to be performed would be to re-factor the code to use an open source equivalent at some point. However this was not considered essential at the beginning of the project. As it later turned out this would not have been an issue since BEA subsequently acquired SolarMetric and announced plans to open source it in 2006 (*Mountjoy, J. 2006*) as the Apache OpenJPA project (*Apache 2007b*).

A number of use cases were proposed which would be worked on following the launch of NGOSSoil with a view to helping the wider community to understand how the technology could be used. One of the projects looked at was an integration of the NGOSSoil into OpenNMS using OSS/J to provide an NGOSS modelling capability.

A number of parties were found who were willing to commit resources to the project and who had varying motivations for getting involved as illustrated in Table 30. However Nortel's management would not commit resources or release the code unless at least two developers could be found from the service providers. Unfortunately the project foundered because, despite presentations to the TMForum's Service Provider Leadership Council (SPLC), no service providers (including BT) were prepared to make this level commitment.

Table 30 Analysis framework NGOSSoil Project Analysis Using 6 W's
(After Chapman, C. B. & Ward, S. 2003 page 4)

Who (parties)		Why (motives) What do the parties want to achieve?	What (design) What is it that the parties are interested in?	Which way (activities) How is it to be done?	Wherewithal (resources) What resources are required?	When (timetable) When does it have to be done?
SP	BT	OSS cost reduction Less Integration Tax NGOSS realisation	Encourage ISV's and other SP's to investigate open source	Leadership of TMForum open source program	1 Senior research engineer as leader of open source program	initially 1 year from April 2006
ISV's	Nortel	OSS cost reduction Less Integration Tax Address 3Gpp, DMTF and NGOSS standards Standardisation of commodity infrastructure	Promote a single open source model mediation framework to reduce cost of mediation for all equipment vendors Create an option to later imbed code in products	Create an open source project around an existing technology developed using community source within Nortel. Only possible if service provider offers developers	1 project manager promised 2 developers promised	initially 1 year from April 2006
Integrators	Invocom	Build Integrator / Consultancy business	Build Invocom brand awareness Demonstrate thought leadership through EngD sponsorship	Fund Action Research into Open source	Sponsorship of EngD student	NGOSSoil as a tactical project within 4 year EngD research
	Cognizant	Build Integrator / Consultancy business	Opportunity to showcase Indian Open Source practice	Be seen as active contributors to an open source OSS project Provide integration expertise	5 Indian developers promised	initially 1 year from April 2006
Universities	UoS,	Research opportunities Student projects Funding	Build relationships with the Telecoms industry through the TMForum	Eng D action research Contribute undergrad projects Bid for collaborative funding	1 Research Engineer potential research growth Living lab Facilities Support for TMForum gForge site	initially 1 year from April 2006
Trade Assn.	TMForum	Demonstrate value to members through innovative projects and realisable standards	Demonstrate practicality of NGOSS Enhance Catalyst program Explore open source Keep ISV's / SP's happy	Practical development of recognised open source program Agree open source strategy at board level	Waive participation fee for active Universities Assign 1 staff member to open source program	initially 1 year from April 2006

4.3.4 Evaluating NGOSSoil

The failure to bring the NGOSSoil project to fruition was obviously a great disappointment to the team who had put a great deal of effort into developing the project plans. However it is useful to analyse what can be learnt from the experience in order to understand what it might take to make a future open source OSS project more successful. In order to do this it is important to understand the potential significance of NGOSSoil to both to the wider OSS community and its major sponsor, Nortel.

4.3.4.1 The Potential Significance of NGOSSoil to the wider OSS community

In order to appreciate the potential value of Nortel's offer for the TMForum's NGOSS program, it is necessary to consider how it compared with the DMTF's WBEM CIM open source offerings. WBEM implementations are now available for most operating systems and there are several proprietary and open source implementations of CIM repositories as illustrated in Table 31. These projects provide central CIM repositories with open API's allowing manufacturers to design their own management providers which map their equipments' state into the common model.

In a comparison with other network management standards, Cisco architects (*Schott, J. & Westerinen, A. 2003*) describe the NGOSS SID as a relatively new standard with a small number of implementations. Certainly there are far fewer implementations of the SID than the DMTF CIM. A model repository which could hold both CIM and SID models and provided similar capabilities to most WBEM implementations which allow the plugging in of device adaptors would certainly be very useful in promoting NGOSS to the wider community.

In a service providers' review of management standards, (*Schoenwaelder, J. 2003*) notes that the main practical use of CIM schemas today seems to be the definition of data structures used internally by management systems. These CIM repositories appear to be used primarily by device manufacturers to provide structured configuration and management databases internally to their devices - even though this is exposed to external configuration and management tools using CLI or SNMP. This is certainly a use described by one Nortel architect (*Hobbs, C. 2004*) and in Microsoft Windows Management Instrumentation (WMI) documentation *(Microsoft 2007)*.

The main draw back with the current CIM specification is that although it is object orientated, it does not use UML for model definition. Instead the specification is rendered visually as MS Visio diagrams (which are similar in appearance to UML), each having separate model specifications encoded using a DMTF proprietary standard called Managed Object Format (MOF) (*DMTF 2007b*) (This is not to be confused with the OMG's Meta Object Facility used for UML (*OMG 2002*)). Keeping the diagrams and the MOF specifications correctly aligned is an overhead for the DMTF which would be avoided if UML was used instead.

Table 31 Common Information Model (CIM) Repository Implementations	
Project Name	Details
SBLIM	Sponsored by IBM. 'Standards Based Linux Instrumentation for Manageability'. The goal of this project is to provide a complete Open Source implementation of a WBEM-based management solution for Linux. (*SBLIM 2007*)
OpenWBEM	Sponsored by Quest Software and Novell - a complete implementation of WBEM. Runs on Linux, Solaris, HP-UX, AIX, Mac OS X and Novell Netware (*OpenWBEM 2007*)
OpenDRIM	Sponsored by Chinese and Asian SW developers. 'Open Distributed Resources Information Management'. Goal is to develop an Open Source, standard-based, centralized Linux resources management system (*OpenDRIM 2007*)
WBEMServices	Sponsored by Sun. An open-source Java™ implementation of WBEM The APIs are based on Java Specification Request (JSR) 48 (Note appears to have had little work since 2004 but Java JSR48 work is going on in OpenDIRM and SBLIM) (*WBEMservices 2006*)
OpenPegasus	Sponsored by the Open Group. Pegasus is an open-source implementation of the DMTF CIM and WBEM standards. Builds and runs today on most versions of UNIX(R), Linux, and Microsoft Windows. (*TheOpenGroup 2007a*)
Windows WMI	Windows Management Instrumentation (WMI) is the Microsoft implementation of Web-based Enterprise Management (WBEM) it comes as standard with most windows platforms. (*Microsoft 2007*)

CIM repositories are configured by loading them with device specific MOF files which extend the core CIM model to provide a proprietary model of the device being managed. Cisco has proposed that future releases of the CIM be specified in UML but given the growing installed base of CIM repositories and the enterprise industry's adoption of the current standard, it will be a difficult task to persuade DMTF members to change the modelling language used for future releases of the CIM standard unless some bridging technology is made available which can do as good a job of storing model information as the existing CIM repositories. The existing specification format will probably continue to be used and developed for some time.

Manufacturers such as CISCO and Nortel, who serve both the telecommunications and enterprise markets, have committed to service providers that they will implement the TMForum's MTOSI interface (*TMForum 2005f*) which supports a model based upon the SID for network configuration. They want to align the SID and the CIM in order to avoid having to implement more than one management repository for future network devices.

The use of UML rather than MOF offers the possibility of automatic code generation using MDA techniques for both the model and the interfaces. Limited code generation using AndroMDA (*AndroMDA 2007*) has been demonstrated by the Cisco sponsored ModelWizard project (*Cisco 2007*). ModelWizard is a set of Eclipse plug-ins that support Information Modelling using The Eclipse Modelling Framework (EMF) (*Eclipse 2007b*) and UML version 2. The plug-ins, although initially targeted at the DMTF's CIM, should support any Object Orientated UML class model.

However there is as yet no open source repository which can instantiate CIM and SID models specified in UML and which also provides the ability to integrate management provider plug-ins with a similar degree of capability to the WBEM implementations based on MOF. Nortel's NGOSSoil proposed to fill this gap with a model repository which was not tied to the MOF format and which could be configured using XML configuration files which could be translated from UML's XMI format using XSLT style sheets.

4.3.4.2 The Potential Significance of NGOSSoil to Nortel

We have see in Section 4.3.4.1 that both the TMForum and DMTF communities stood to gain from the proposed NGOSSoil functionality. However the plan to extend this model to the wider OSS community could also have provided additional benefits to Nortel.

Nortel had prior technical and business experiences which made them favourable to the potential benefits of open source in helping with OSS development. As discussed in Section 1.3.2, one of the major costs which Nortel faced in common with other equipment vendors was the cost of integration into many different customers' OSS systems. They had a history of actively participating in standardisation efforts in order to reduce this cost and were actively involved in both OSS/J and MTOSI. Section 1.3.2 noted that Nortel had a prior history of developing a technology in the 1990's called Integrated Network Management (INM). INM was based on the TMForum's GB-909/TINA-C standard for CORBA building blocks and was supposed to deliver a plug and play architecture which would allow Nortel to integrate easily with customers' OSS systems based on similar standards. Unfortunately the OSS industry never really managed to create a compatible set of CORBA building blocks in the OSS layer and this had limited the external technical and business benefit which Nortel could derive from pursuing the standards.

Section 3.4.3.1 considered how during an OSS acquisition strategy, INM had been re-branded as 'Preside' and the INM team had become a separate Nortel business unit which had acquired a number of OSS vendors in a bid to create a portfolio of OSS products. Within Nortel it had been hoped that these standards would allow Nortel to develop consistent OSS management capabilities for products developed across the corporation. Even if it was difficult to influence the externalities in the OSS market in order to maximise the benefit from standards compliance, Nortel could still use the standards to maximise the integration benefit customers derived from its own portfolio of OSS products.

However this strategy was thwarted both by technical and business issues. Firstly, the creation of a separate 'Preside' business unit made it difficult to propagate the technology across the corporation since other lines of business who wanted to use the core Preside technology had to pay for support and could not recognise the revenue from the product. Secondly, the first generation CORBA building blocks had been originally designed for managing Sonet/SDH networks and were never really suitable for managing other technologies which required bulk data transfer and more sophisticated alarm and event management.

Potential adopters of the technology within the corporation were faced with a dilemma. Although, for the good of the corporation, they would like to support Preside, they felt little control or ownership of the technology roadmap and perceived that the core developers did not have an appreciation of their particular technology needs. This of course was further complicated by the problems which the CORBA and TINA-C standards were having on which these components had been modelled (as discussed in Section 1.4). The revenue recognition

issues just made it more difficult to justify any investments. As a result, and perversely, it was sometimes easier for business units to develop their local solutions or even go outside the corporation for alternative products.

These problems persisted until following the collapse of Nortel stock, most of the Preside OSS assets were divested and Nortel was no longer in the OSS business (Preside remains as a brand but not as a profit centre). However Nortel had learnt a number of lessons out of the Preside experience when they came to design a second generation technology to manage converged 3GPP networks called the Multi Service Platform (MSP).

Firstly, from an organisational perspective, although a core technology group reporting into the CTO's office would foster the development of the platform, this would not be a profit centre and they would not be the only developers of the technology. The core management platform was to be developed under a 'Corporate Source' model rather than as a platform supported by a single business unit. Any development community wishing to investigate or leverage the code could commit as much or as little resource as they felt they needed without a complex cross-charging arrangement. This would give the lines of business much more local control over the risk and return on their investment, a sense of ownership of the technology and the ability to adapt it to their needs.

Secondly Nortel tried from the start to make the system 'model agnostic' so that it could be used for multiple technologies. It could accommodate Enterprise DMTF and Service Provider NGOSS models and would be capable of supporting MTOSI and OSS/J interfaces.

Thirdly, using a corporate source model gave the corporation more flexibility than the essentially waterfall process under which INM was originally developed. When INM was first created, Nortel's NPI process synchronised the marketing and R&D functions using a 'Stage Gate process' which was linked with the budgeting process and approved the staged allocation of resources for new product introduction as discussed in Section 3.6.1. At each stage of development every program had to have identified at an increasing level of certainty a potential market, lead customers, a set of features and an estimate of the development costs before approval was given to proceed. Latterly, Nortel had tried to make the process less rigid but the corporate source model potentially allowed for even more internal flexibility and for innovation by R&D directors outside of the NPI process.

Thus corporate source had the potential to overcome internal communication and market barriers to acceptance of a technology direction set by the CTO's office. It might provide a means for technology to diffuse through the corporation and be implemented by lines of business as the market opportunity arose. As discussed in Section 3.4.3.1, the corporate source model was being experimented with by a number of equipment vendors such as Lucent and Alcatel with similar benefits.

Discussions with Steve Jerman[41], the maintainer of Cisco's ModelWizard project, revealed that one of the benefits to Cisco of creating the ModelWizard project as open source rather than just corporate source was in influencing opinion inside and outside the Corporation. It nailed Cisco's colours to the mast in front of customers and competitors. Cisco's technology architecture group were developing an internal information model (called Chameleon) combining the SID and CIM in a single UML model which they were proposing to the DMTF as the basis of CIM v3. The ModelWizard showed how a UML model of the CIM could be realised as code and Steve Jerman's group were using it both externally to influence the DMTF standards community and internally to influence the developers across Cisco's lines of business to use a single model.

Similarly, Nortel's idea of committing part of the MSP technology to open source could create a much stronger means for the technology owners to influence its adoption across the corporation. If it could be seen that a wider community of customers, ISV's and even competitors were contributing to the technology, it would be much harder for internal forces to resist its introduction.

4.3.4.3 Lessons Learnt from NGOSSoil

It can be argued that NGOSSoil was the nearest thing to an NGOSS component that had yet been proposed to the community. The developers and architects within Nortel who presented the DML to the TMForum community clearly believed in the technology and saw it as a route for practical implementation of a number of NGOSS and WBEM standards. Nortel developers had used some of the DMTF's open source repositories in products and understood the value of community development[42].

We were told that a number of the developers who had been on the MSP project were already quite committed to the ideas of open source - and might even contribute in their spare time. A number of these individuals also participated in the standards activities and could see how it could help their work. Their view was that anybody needing to implement an NGOSS application and who understood the issues of creating an OSS would come to see the value of the component. They were clearly very proud of their work and wanted to see it taken up and actively used. If the DML had any technical drawbacks, it was unlikely that these would have been critical impediments to its use. The enthusiasm of the developers for the open source project and consequent disappointment when it failed to get corporate support demonstrated the reality of the issues concerning the differences between a development and marketing mindset discussed in Section 3.6.1.

A number of OSS component vendors did take the time to review Nortel's offer at the Team Action Week but none were prepared to commit resources. One can only surmise that a

---

[41] Q&A session with Steve Jerman, Cisco, at DMTF WBEM Developers Conference Santa Clara CA 8/12/2005

[42] Private conversation with Chris Hobbs at DMTF WBEM Developers Conference Santa Clara CA, 5-8/12/2005 , Chris Hobbs had used Open Pegasus for product development in Nortel and written a book on WBEM (*Hobbs, C. 2004*)

number of them already had similar (albeit first generation) components already at the heart of their OSS offerings or a similar component to Nortel's under construction and would never be able to make a short term business case to substitute an external capability. On the other hand a system Integrator, Cognizant, was prepared to commit resources as they could clearly see the value of the component for integration.

Although one will never know all of the reasons the project did not go further, on reflection, several are worth considering.

Firstly the NGOSSoil component required complimentary products and services in order to be useful. The component naturally sat at the heart of an NGOSS environment but would not find application in production environments until NGOSS became more of a reality. Unless someone was implementing an NGOSS-like solution, they would find it hard to build a conventional business case to invest heavily in the component. The early adopters would only ever be researchers or developers who were in the early stages of developing an NGOSS system. Although the component had the potential to solve a number of integration problems, some real use cases would need to be demonstrated before late adopters would be attracted to the project. If the component went into production, then it would probably need commercial support from the project and it was not clear how this would be provided.

Secondly, the service providers were clearly not ready for an open source model of OSS production. Nortel wanted an indication from the service providers of their interest and received none. Having divested themselves of R&D, the service providers no longer had any mechanism to participate in open source projects - except where staff in the operations part of the business could justify an involvement 'below management's radar'. Even when a number of OSS suppliers were beginning to embrace the open source model, the service providers were not yet in a position to actively participate as part of their procurement strategy.

Thirdly, Nortel put preconditions of service provider involvement before releasing their code. From the discussion above the service providers were unlikely to be able to contribute while the likely contributors would have been other equipment vendors or solutions integrators. As the authors of the code, Nortel had most to gain if it was taken up by the industry but releasing the code would have required some relatively small but strategic investment on their part. If they believed in the open source model, then needed to make a sufficiently long term commitment to it and this decision had clearly not been made within the corporation at a senior level.

Nortel's own developers were a key initial resource who were needed for the project and who were likely to volunteer additional time to keep it going. The fact that Nortel weren't prepared to first make the initial investment of their own people made it very difficult for others to commit the same level of investment. If Nortel had made the code and documentation available before asking the TMForum members to make a commitment, it is possible that a

number of the members would have taken the time required to evaluate the project and later committed to get involved.

Finally, the TMForum may attract the wrong community. This became very apparent when in 2005 the researcher visited the DMTF developer conference in San Francisco[43]. This is actively aimed at developers implementing DMTF standards. The level of technical discussion was markedly different to that at the TMForum which tends to attract senior managers and solutions architects. Although it was suggested in Section 3.4.2.2 that there was still potentially a very large developer community across the service providers who might be willing to leverage open source OSS components, most participants in TMForum activities are not close enough to the coal face in their organisations to get involved in coding. Very few real programmers attend TMForum events and so any open source program would first have to find a way to reach out directly to the developer community beyond TMForum attendees.

4.4 Related Action Research

As discussed in Chapter 2, Action Research has had limited application within the Telecommunications sector. However as part of the evaluation it is useful to contrast some related action research projects with OpenOSS.

4.4.1.1 R&D Net

The R&D Net project 'Knowledge sharing and creation in networked R&D processes in the telecommunication industry' (2000-2003) was conducted by the SimLab multidisciplinary research unit at Helsinki University of Technology (*Hirvensalo, A., Evokari, J., Feller, J., Pekkola, K., Turunen, P., & Smeds, R. 2003*). Eight MSc students researched various aspects of collaborative R&D conducted between Nokia, TeliaSonera (a Swedish/Norwegian network operator), and Elektrobit (an electronics design house). In contrast with the OpenOSS work, they were looking at collaboration across established (contractually defined) business partnerships rather than within the much less structured environment of a trade/standards organisation like the TMForum or an open source project.

The Action Research methodology rigorously followed SimLab's Business Process Simulation Method and was facilitated through workshops with groups of around 25 R&D professionals drawn from the participating organisations The project used Action Research to study knowledge sharing and creation in networked R&D processes in the telecommunications industry and was motivated by the work of (*Amidon Rogers, D. M.*) who suggested that in the future R&D will be characterised by a fundamental shift toward virtual, networked organisations that transcend conventional boundaries and will be conducted by Self Managing Knowledge Workers.

This analysis obviously has an affinity with the issues raised in Section 3.6 which considered the potential value of open source in collaborative working. However the focus of the Action

Research was on helping the partners through business process improvement of their collaborative R&D processes and the researchers did not actively participate in the development work of the R&D organisations. Some research was done into the use of ICT to facilitate collaboration between R&D organisations but no explicit research was done into open or community source, which is surprising since this industry is software intensive.

4.4.1.2 University of Delft UMTS Test Bed

More akin to the OpenOSS work, (*van de Kar , E. A. M. 2005*) has reported an Action Research project conducted with T-Mobile to develop a year long UMTS test bed at the University of Delft. Researchers at the University actively engaged the industry and used personal contacts to find a group of partners in order to create a representative supply chain (or 'Value Network') around a trial UMTS service. This included the Network Operator, Handset supplier (Nokia), Content providers, Software suppliers and students at the University of Delft. Like the OpenOSS project, each partner had a different reason for participating; T-Mobile used the project to showcase public advantages of UMTS and the University used the test-bed for concept exploration, prototyping and testing of new services.

Unfortunately, the researchers share very little detail as to the Action Research methodology used and do not refer to other Action Research literature. However their methodology appeared to have an 'analysis stage' determining the issues and theories driving the action research and an 'action planning stage' when the project was initiated. They had active participation by the researchers in developing the project and a concluding survey of the service users and the participating firms to evaluate the value of the experience although an intended business analysis was not completed by the project.

It appears that the objectives of this work were quite similar to the objectives of the OpenOSS work in the TMForum; establish a living lab where partners can learn together about the issues of introducing a new telecoms service. Whereas the Delft work was service creation focused, this work was focused on service management. The key advantage which the OpenOSS team experienced by working within the TMForum was a ready-made mechanism for attracting partners and disseminating results. The TMForum is planning to introduce a 'service prototyping' track to the Catalyst program starting at TMW Dallas 2007[44] which will encourage similar service creation exercises to that promoted at Delft. In the future this may provide opportunities for Universities to participate in combined service and service management demonstrations.

4.4.1.3 Process Portals

Appendix D.1 describes how Sun persuaded the OpenOSS Catalyst to include a portal server in the architecture. A portal server allows individual application vendors to decouple their web

---

[43] DMTF WBEM Developers Conference Santa Clara CA, 5-8/12/2005

[44] Private Interview with Catalyst program Director Debbie Burket conducted at TMWorld Nice 2006

GUI technology from their server technology. If the web GUI is delivered as JARs defining a portlet according to the JSR 168 Java Portlet specification (*JCP 2003b*), the GUI can run alongside GUIs from other applications within a common portal server. The portal server allows centralised administration of user identity, the applications which a user can access and individual user control of the layout of their 'desktop' of portlet applications.

(*Puschmann, T. & Alt, R. 2004*) have conducted an extensive Action Research project looking at the architectural implications and the practical uses which can be made of portal servers for intra-organisational and inter-organisational integration. The research was conducted with 9 companies including Deutsche Telekom, DaimlerChrysler, Hewlett-Packard, Bosch and SAP.

The focus of the OpenOSS prototype was on simply investigating the usability of open source portal servers as part of an OSS architecture. (*Puschmann, T. et al. 2004*) used web services and EAI adaptors as the interface from their portlets back to their applications. In OpenOSS OSS/J was used, but this work could be extended to use the forthcoming OSS/J web services profiles. The conclusions of the OpenOSS work were simply that using standards compliant open source portal servers were a viable strategy for integrating OSS products. However Puschmann's work illustrates that the Action Research around OSS portals could be taken considerably further in a project looking at the process flows and UI requirements for a telecommunications organisation.

4.4.1.4 Action Research as Entrepreneurship

One characteristic in common between the above projects and the OpenOSS project was the need to establish the partnerships in order for the work to proceed. Section 2.1.4 discussed the issues of obtaining access to a situation. Whereas researchers in the R&D Net project gained access to a pre-existing business network, the researchers in the Delft UMTS test bed needed to create business interest from a set of partners to encourage participation in their project.

Similarly much of the early work in the OpenOSS project concerned creating partnerships both for the Catalyst and for NGOSSoil. These activates are much akin to what is often referred to as business development. (*Rasmussen, B. & Nielsen, T. 2004*) have established that there is a natural affinity between business entrepreneurship and the Action Research process, particularly in the SME environment.

If, as discussed in Section 3.6, industrial innovation is increasingly happening at the edge of organisations, the opportunities for individual knowledge workers, SME's and academic research groups to work with industry will increase however researchers will have to develop more entrepreneurial skills in order to get involved. Most Universities now have business development units tasked with both attracting industrially sponsorship and commercialising research results. However these functions presently appear to sit uneasily alongside the research faculty; one group motivated by commercial interests, the other by peer reviewed publication. Often faculty get no recognition for the work required to establish a useful project.

The OpenOSS project has demonstrated that Action Research does appear to offer a research paradigm which provides a formal mechanism for academic research to be recognised which entrepreneurially engages with the Telecommunications industry.

Chapter 5 Intervention: Digital TV Service Management

Chapter 5 Intervention: Digital TV Service Management

5.1 Introduction

In the OpenOSS Catalyst discussed in Chapter 4, the team proved that it was feasible to simulate aspects of a next generation network and create a rudimentary NGOSS solution using open source components. However the scope of the investigation had been limited due to time and resources. The natural next step in the EngD research program was to find a more sophisticated network management problem and see if NGOSS design principles could be applied to creating an OSS demonstrator which would be built using open source components.

Discussions with Arqiva over an extended period eventually led to the opportunity to create a proof of concept for managing SLA's in a digital TV (DVB-T) network. This work was used by Arqiva to inform a tender specification for an OSS system to manage the forthcoming roll out of Digital TV as a replacement for the UHF services in the UK. This represented a significantly innovative requirement for Arqiva as they had no previous experience of applying OSS technology to managing a broadcast network and had limited internal expertise in OSS systems design.

The Diagnosis phase of the project was initiated at a meeting on 10/08/2005 which agreed the terms of reference for the engagement and allowed the researcher access to Arqiva's premises. Between 10/08/2005 and 27/1/2006 a series of meetings were held with key Arqiva Stakeholders and an Action Plan was developed consisting of a proposed trial which addressed the aspects of DVB-T service management which Arqiva then considered to represent the most risk to their program.

The Action Taking phase began with a meeting between the key trial stakeholders on 27/1/2006. This resulted in actual trial development and integration leading to an initial demonstration on 28/4/2006.

At this point the project entered an Evaluation phase during which time the demonstrator was shown to key Arqiva stakeholders in order to obtain feedback. At the same time an RFQ was prepared which was informed by the results of the trial. This formed the basis of a tendering process inviting potential suppliers to bid to develop an OSS system for the DVB-T network. Suppliers tendering were invited to view the demonstration in order to better understand Arqiva's requirements. The proof of concept continued to be developed beyond the initial demonstration and the resulting OSS/J interfaces were contributed to the OpenNMS project as described in Chapter 6.

5.2 Action Research and the NGOSS Lifecycle

The TMForum recommends using a lifecycle methodology for developing an NGOSS system (*TMForum 2004a*) which draws on RM-ODP (*ITU-T 1997b*), OMG's MDA (*OMG 2007b*) and the Zachman framework (*Zachman, J. A. 1987*), (*Sowa, J. et al. 1992*). The methodology

guides the designer through a process of defining Business, System, Implementation and Deployment views of the NGOSS system using a cyclical process called SANRR (Scope, Analyse, Normalise, Rationalise, Rectify) outlined in Table 32

Table 32 NGOSS Lifecycle SANRR Methodology	
Scope:	define solution boundary including solution mission, goals and business high level use case
Analyse:	Document existing (legacy) and desired environments with detailed use cases, process maps and policy lists
Normalise:	Map current view onto common vocabulary to achieve a 'single unified model'
Rationalise:	Examine the normalised model for gaps
Rectify:	Modify, delete or add functionality identified in rationalise stage

The NGOSS lifecycle represents a set of guidelines which were mapped broadly onto the Action Research process as follows;

During the Diagnosis phase the problem was Scoped and Analysed in order to understand DVB-T SLA management problem.

During the Action Planning phase, a process of Normalise, Rationalise and Rectify was used to create an NGOSS Business View of the desired functionality using a UML information model based upon the SID along with an NGOSS System View based upon the TAM.

During the Action Taking phase, a subset of the NGOSS System View was selected for implementation. The output from the Action Taking phase was an NGOSS Implementation View realising some aspects of the System View by combining COTS and Open Source components using OSS/J. Since the system was intended for a trial, the NGOSS Deployment View consisted of a set of test scenarios to demonstrate the capability of the technology. The scenarios defined the trial network model and data fill, the alarm correlation rules (policy), and a network alarm simulation based upon the output from real DVB-T network equipment.

5.3 Diagnosis

In order to understand the problem facing Arqiva, it is necessary to consider how the DVB-T service would affect their business.

5.3.1 Industry Background

Since the early days of television, the dominant broadcast delivery mechanism, at least in the United Kingdom, has been via terrestrial transmitters. At the time of the investigation two Service Providers operated the UK's terrestrial TV networks on behalf of the major broadcasters; Arqiva Communications (*Arqiva 2007*) and National Grid Wireless (*NGW 2007*)[45]. These companies compete with each other and with Satellite and Cable Service Providers to offer mass distribution services for the UK broadcasters.

In 1996 the UK government passed legislation which provided for the introduction of a national digital terrestrial television (DVB-T (*ETSI 2004*)) which launched on 15th November 1998 as a low power service running in parallel with the main PAL terrestrial broadcasts (originally called ONDigital now called Freeview (*Freeview 2007*)).

[45] At the time of writing, August 2007, there is the possibility of these two companies merging.

Despite the availability of multi-channel satellite, cable and more recently low power digital services, the analogue UHF PAL service is still the broadcast medium received by 55% of the 60 million TV sets in the UK (*OFCOM 2006*). However in harmony with administrations across Europe, the UK Government recently announced their intention to switch off the analogue TV network and replace it with a high power DVB-T network in a rolling program beginning in 2008 and completing in 2012. This will be a significant undertaking requiring the phased transition of millions of households to digital TV reception either through the use of external digital set top boxes or new TV receivers which natively receive DVB signals.

The digital switchover program represents both a significant opportunity and a threat to the terrestrial broadcast service providers. Inertia has kept many viewers in the UK using UHF TV but the switch over program will create a market discontinuity which will force all consumers to change their receiver technology and in so doing possibly replace their broadcast TV reception with an alternative transport medium. The program comes at a time when an increasing number of alternative video delivery technologies are already available to rival the dominance of terrestrial transmissions as the preferred TV medium in the UK. Although 73.3% households in the UK receive digital television services on their primary TV, only 22.5% use the low power Digital Terrestrial services, the balance using well established cable and satellite services. New technologies are also being introduced such as British Telecom's recently launched DSL IP-TV service, BT Vision (*BT 2007b*). Other delivery technologies such as DVB-H (*DVB-H 2007*), 3G and Mobile TV over DAB (*ETSI 2006a;ETSI 2006b*) are about to be launched.

The digital switch-over program is only the latest in a series of significant changes within the broadcasting industry. Where once broadcasting was a vertically integrated industry in which an oligopoly of few broadcasters controlled both the means of program production and distribution, liberalised markets combined with an emerging plethora of content delivery mechanisms have creating a complex supply chain from producer to consumer. Increasingly independent production companies produce programmes which are aggregated into schedules offered through branded channels (such as BBC, ITV and Sky) and made available over cable, satellite and terrestrial broadcast delivery platforms. Margins are under increasing pressure as advertising revenue is being spread ever thinner across multiple outlets. (For instance Google is now predicted to attract a bigger share of advertising revenue than the UK's premier commercial TV service, ITV (*FT Online 2006*)).

Against this background, the principal marketing benefit of the terrestrial DVB-T platform will be as the consumer's least cost option. A one-off purchase and simple installation of a DVB-T box between their UHF aerial and their existing TV will provide a viewer with at least 30 free to air TV channels and 20 radio services. At present market rates this is less costly and less disruptive for the consumer than a subscription to cable, satellite or IP-TV. However, the economics for the broadcasters are less clear. While broadcasting remains significantly less

capital intensive than per-subscriber cable or DSL IP-TV provision, the capital cost of replacing the UK's UHF transmitters will be very high. The DVB-T Service Providers will have to show a 25 year return on investment while charging the broadcasters access rates which are competitive against the already operational digital cable and satellite services. Furthermore the terrestrial broadcast services will have to remain compelling to consumers when compared with the interactive offerings from IP-TV and digital cable.

The successful migration of consumers to the DVB-T platform will be critical for all of its stakeholders. The transition process will occasion very high levels of public scrutiny through the UK government regulator, OFCOM. The terrestrial broadcast service providers will want to retain the loyalty of the majority of the UK population by offering the most cost effective and reliable TV delivery mechanism having a service quality which is as good or exceeds the current analogue services. At the same time the aforementioned economic pressures will force the terrestrial service providers towards a much leaner operation while the network being managed will be significantly more complex. Furthermore, whereas the analogue TV networks were adequately run as best effort exercises, new DVB-T Service Level Agreements (SLA's) currently being individually negotiated with each of the broadcasters sharing the DVB-T platform will require that the Service Providers are able to tightly monitor their performance in real time and offer service credits against each SLA breach.

5.3.2 Managing SLA's in Telecommunications and Broadcasting

The nature of the telecommunications industry was discussed in detail in Chapter 3. This section will extend the discussion to the broadcasting industry.

To date the broadcasting and telecommunications industries have coexisted but only tangentially worked together in the development of common standards or processes for running their businesses. For instance, until 1993 the BBC's contribution network, although provided by BT was entirely analogue and independently managed from BT's core services. When they switched providers in 1993 to Energis, who for the first time carried their vision signal using an SDH network which integrated video with telecommunications traffic (*Russel, D. R. & English, A. R. 1995*), the basic processes for managing the vision service remained substantially the same. (The researcher was a project manager for the network management aspects of this transition).

Traditionally broadcasters have relied on a relatively small number of transmitters to reach a mass population with high bandwidth TV services. The analogue services were characterised by a one to one relationship between each transmitter and the service it carried and so the service impacts of faults were easy to determine using unsophisticated Supervisory Control And Data Acquisition (SCADA) solutions. The Service Providers' interactions with the broadcasting customers were through close, professional relationships based on trust

established between the transmitter control rooms on one side and the presentation controllers on the other.

In contrast with broadcasting, telecommunications service providers have learned how to serve a complex mix of domestic and high value business customers sharing the same network. They could only achieve economies of scale if common network equipment (resources) were used to offer many different services to many different customer types. This led them to develop OSS solutions which allowed them to manage the maintenance of the network resources separately from managing the customer facing service level agreements. High value customers were offered tailored processes for problem escalation and service management even though their service might be sharing common resources with other high value customers or with low value mass market customers. As discussed in Section 3.4.2.1 deregulation in the telecommunications industry also led to a complex market where end user services may be constructed using the services of multiple service providers and so systems and processes were developed for managing complex service supply chains with very low margins.

(*Hayler, K. 1999*) and (*Hibbert, C. 1999*) provide a description of the design and the issues which had to be overcome when building the low power DVB-T Freeview network. Although the network was limited to 81 stations with no down stream dependencies, Arqiva found that the digital service introduced more complexity into the management problem. The existing SCADA system could be extended to manage the transmitters but the SLA reporting requirements required Arqiva to allocate a man week to prepare reports each month. The existing operational model would clearly not be sustainable given the network complexity and SLA demands being placed on the new DVB-T platform.

Thus the terrestrial service providers are being forced to think about managing their broadcast networks using techniques developed for lean telecommunications operators. Designing these capabilities required a comprehensive process and information model such as that represented by the NGOSS eTOM and SID. To be competitive in the long term the DVB-T services needed to be offered under a least cost model and an important consideration was the cost effective use of SLA management technology.

Telecommunications OSS solutions have historically attracted high price tags, in part because of service providers' requirements for scalability and reliability but also because network management represented a small market place where most of the customers had deep pockets. Whereas in telecoms OSS costs could be amortised across a large number of customers, the same solution would be out of reach of most broadcast operators. In summary, the broadcasters were now looking for SLA management solutions based on NGOSS principles but at considerably lower cost than the telecoms industry was used to paying for OSS.

5.3.3 Arqiva's Requirements

Arqiva particularly wanted to address three problems.

Firstly, Arqiva wanted to gain practical experience of highly innovative OSS solutions to managing a DVB-T network. Arqiva had previously participated in a trial of DVB-H technology with their partner, mobile operator mmO2. Publicly the trial was deemed a success (*Short, M. 2006*) but the project had made Arqiva aware that their SCADA systems were only able to present faults and not information about the services affected. The information presented was too closely tied to Arqiva's network and not easily usable by the customer service centre in mmO2. Arqiva needed to understand how it could meet its DVB-T SLA obligations and communicate effectively with its telecommunications partners. This pointed to the need for a trial which demonstrated the use of telecommunications standards such as NGOSS.

Secondly, Arqiva needed to contain the cost of the trial and investigate low cost technology alternatives to a traditional OSS stack. The broadcasting business did not generate the revenues present in telecommunications and Arqiva's business case could not justify the cost of a classical OSS stack. Arqiva needed to investigate more cost effective solutions. In addition, at the time of the trial, Arqiva were in a bidding situation and had not yet won any contracts for the DVB-T roll-out. Arqiva needed to conduct the trial at limited cost without prejudicing the final selection process for a solution if the project went ahead. The trial would inform an RFQ process later in the year. For these reasons an open source based trial was attractive. Not only was it low cost but it would be available for potential OSS bidders to view in order to better understand Arqiva's requirements.

Thirdly, the engineers close to the problem in Arqiva needed to educate their own organisation as to the problems of managing a large multi service network. It was felt that while some people in the organisation understood the difficulty of the task, others had simply assumed that rolling out the new network would be 'more of the same'. This was an opportunity to begin a process of organisational learning as part of the planning process.

5.4 Action Planning

In the action planning stage it was necessary to analyse carefully the nature of the DVB-T management problem in the context of emerging OSS standards. This analysis was done by the researcher working at Arqiva over a number of months and is covered in the following sections.

In addition in the action planning stage it was necessary to build a solution team to conduct the trial. The researcher worked with Arqiva to broker a solution involving contributions from several parties. Invocom and Sidonis offered experience in alarm correlation and a product which Sidonis were developing called StateWise (*Sidonis 2007*). Invocom were contracted to write the initial Arqiva RFQ following the trial. Scysis offered help with investigations into the MIB's present in transmitter systems. Consultancy from OpenNMS was arranged since this would be a key alarm management tool in the trial.

5.4.1 Applying Telecommunications SLA Management Standards

This section covers the analysis done by the researcher into the applicability of NGOSS standards to the DVB-T management problem. Subsequently to the trial, the researcher took the core of this analysis to the TMForum Team Action Week in January 2006 and tested the analysis with the SLA management team and the SID team.

5.4.1.1 SLA Definition

For some time the TM Forum has had a working group looking at SLA management which has published a set of recommendations for SLA development. (The SLA Management Handbook (*TMForum 2004e*)). The SLA Handbook seeks to cover the entirety of a customer's interaction with the Service Provider as described by the 'customer experience timeline' including 'pre-sales', 'service ordering', 'in life experience' and 'service cancellation'. BT have reported using a similar approach to that described in the Handbook for analysing the service management problem for broadband and IPTV customers (*Trigger, J. & Harrison, M. 2006*).

While the SLA Handbook does not require that a Service Provider implement NGOSS, the NGOSS SID model has been designed with a view to supporting the SLA definition process.

The SLA Handbook differentiates between two types of SLA measurements; Key Performance Indicators (KPI's), which are essentially raw measurements taken from the network and Key Quality Indicators (KQI's) which are a set of measurements (often derived from the KPI's) which reflect how the service is perceived by the customer. The KQI's are used in the definition of the Service Level Agreement and are negotiated with the customer.

The handbook suggests a 5 step methodology for developing Service level Agreements listed in Table 33. All five of these steps are necessary when developing a full service level agreement but this study focused only on the first three steps since steps 4 and 5 required access to a representative network. The study was also constrained to only consider the 'in life experience' portion of the customer experience time-line, that is, looking only at Service Assurance SLA measurements. The following sections illustrate a how NGOSS along with the recommended SLA development methodology was used to analyse the DVB-T SLA problems.

Table 33 TMForum steps for developing Service level Agreements

1. Identify service scenarios; Customer usage patterns for the service at each Service Access Point (SAP).
2. Analyse the 'customer experience timeline' for service usage.
3. Identify the service topology. This describes in detail the service resources which comprise the end to end topology of the service.
4. Develop a Transaction Matrix. This is essentially an activity diagram outlining how the service responds to user actions or network events (faults). It provides a tool to identify points at which the user could experience dissatisfaction with the service.
5. Develop Key Quality Measurements based upon analysis of the user experience.

5.4.1.2 Identifying DVB-T service scenarios

At the time of the investigation, Arqiva were biding to the UK broadcasters for contracts for DVB-T provision. The contracts were analysed and some key customer requirements for service level management of a DVB-T network were extracted in order to inform the trial.

These are given in Table 34 and are based upon analysis of real broadcasters' requirements without reference to any particular broadcaster.

Note that for the sake of this study the requirements were limited to those concerning the reporting of SLA violations to the Customer who is the Broadcaster using the services of the DVB-T network Service Provider. The case could obviously be extended to include the customer experience of the householder receiving the DVB-T signal. As shall be shown a complex information model is required in order to support these requirements against a DVB-T network.

Table 34 Simplified DVB-T Service Scenarios

Requirement	Description
Self Reporting	Customer should be able to access on-line and in real time information as to the state of the network including current faults and SLA violations
Fault Reporting	System should summarise faults and effect on transmission services within 5 minutes of fault occurring
SLA violation reporting	System should provide tabular reporting of Incidents and Faults categorised by Station, Station Class and Broadcast Service including 'near misses' when a major fault occurs but service was not affected. SLA reporting to include power reductions, un-receivable signal, and incorrect content or service information in the DVB-T multiplex.
Billing Credits	Invoicing to include detailed calculation of total service credits due to SLA violations.
Planned Work	System must support procedures for agreeing planned work and suppressing Fault/SLA violation process when planned work in is progress

5.4.1.3 Identifying DVB-T service topology

Figure 14 provides a simplified view of a DVB-T transmitter network topology. The customers of this network are the broadcasters who target their content (advertising, news, local programming etc.) at specific broadcasting regions. In practice more than one broadcaster may contribute to the content of a DVB-T MUX and the served geographic regions may vary between broadcasters which would require the model to include the capability for dropping and insertion of programs into the multiplex at any point throughout the network. However in practice the number of insertion points is relatively limited and in Figure 14 the simplifying assumption is made that that each DVB-T MUX is pre-assembled by the broadcaster.

The transmitter stations have service level objectives classified in importance according to the population served and their position in the broadcast chain. Platinum stations are fed directly using physically diverse circuits from the broadcaster or from a satellite feed. Gold, Silver and Bronze stations receive their signals off-air from upstream stations and re-broadcast them to their immediate vicinity. For additional protection, stations may also be able to receive and re-transmit out of region programming in the event of complete failure of an up-stream station. Gold, and Silver stations usually demodulate and regenerate the baseband signal while Bronze stations use a simple frequency transposer.

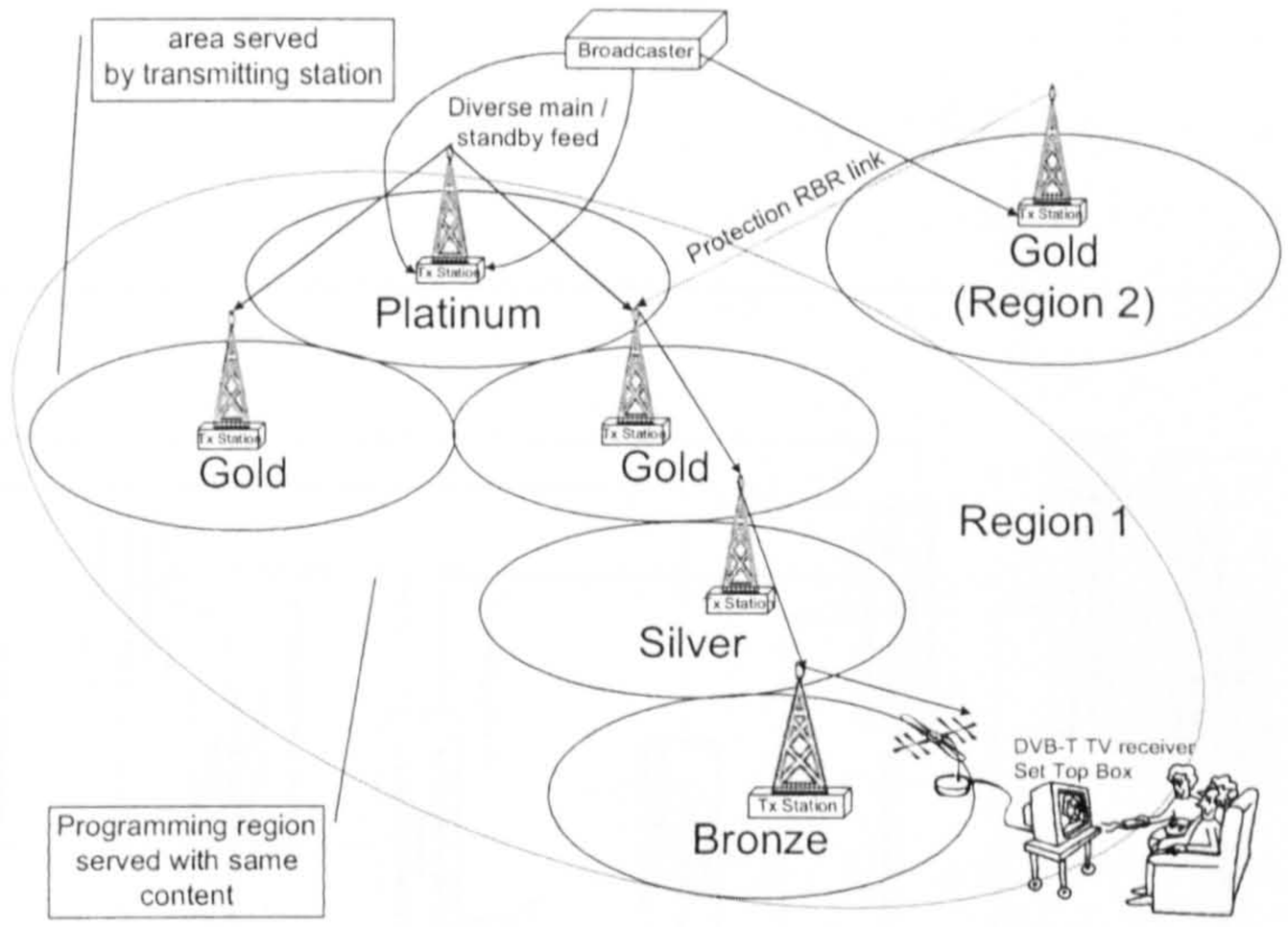


Figure 14 Simplified DVB-T network

It should be emphasised that these requirements are simplified but illustrate the key aspects of the business problem to be solved. The next section will consider how NGOSS can be applied to modelling this service topology.

5.4.1.4 Developing an NGOSS System View of DVB-T Service Management

The NGOSS SID provides a rich telecommunications information model for service level management. The UML diagram in Figure 15 applies key concepts from the SID to a DVB-T network.

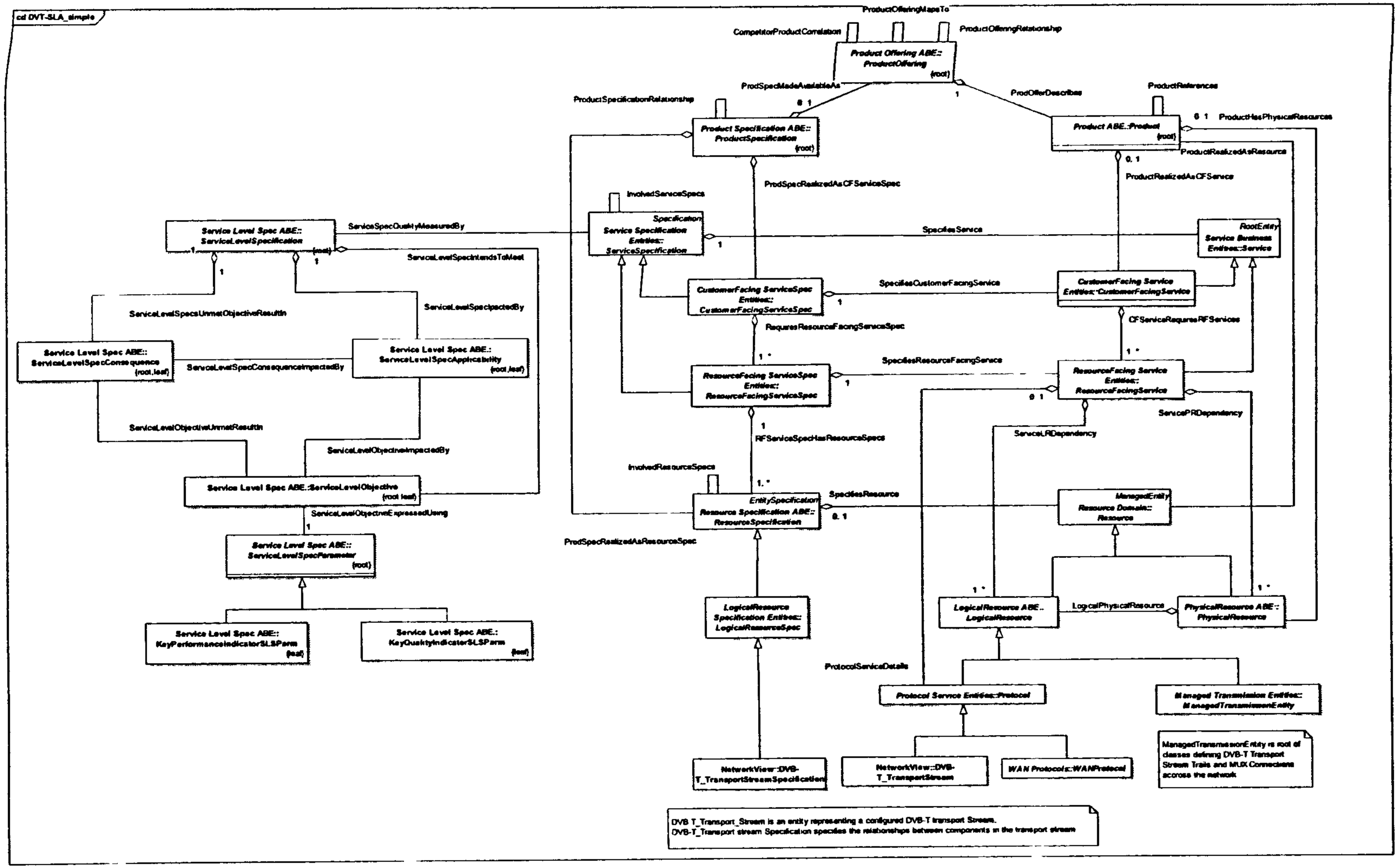


Figure 15 SID Product Service Resource relationships

The SID can appear daunting as it is a complex model which contains numerous Business Entities of interest to a Service Provider. The SID provides a structure for categorising information but it is important to realise that the business entities in the SID are all abstract classes; that is, they provide a starting point but need to be extended for real application. The key to using the SID is to scope the problem - i.e. understand which concepts are applicable to one's problem domain and focus on them without getting swamped by the complexity of the rest of the model.

Central to the SID are the subtle distinctions it makes between Products, Services and Resources (*TMForum 2004g*) which are illustrated on the right hand side of Figure 15 and summarised below.

Business Entities in the Product Domain record among other things, what is offered to the market, how it is priced and also specific instances of Product Offerings sold to specific Customers. Business entities in the Service and Resource domains represent how Products offered to the market are implemented.

A Resource is a component in a Service Providers infrastructure which is used by a Service. For instance a Physical Resource might be a building or a piece of transmission equipment while a Logical Resource might be a circuit carried through the transmission equipment. Services which are only used internally by Service Providers are called Resource Facing Services. Resource Facing Services are implemented using Resources.

A Product is an externally facing representation of a Service procured by the market. Services which are offered externally as products are Customer Facing Services. Customer Facing Services may be constructed from multiple Resource Facing Services. Customer Facing Services are procured by the market but their representation is via a Product Offering.

Products, Resources and Services all have Specifications which describe their invariant attributes. Thus a Service Specification will describe how a particular type of service may be constructed using other services or resources having particular specifications. A Service instance might then be considered as an instance of a Service Specification which also depends upon other instances of Services and Resources.

The left of Figure 15 also shows that each Service Specification can have associated Service Level Specifications. This encapsulates the key concepts from the SLA Management Handbook (*TMForum 2004e*) discussed above. A Service Level Specification is defined by one or more Service Level Objectives which are expressed in terms of Key Performance Indicators and Key Quality Indicators. The ServiceLevelSpecConsequence defines what should happen if the service level spec is breached; for example if there should be a service credit. The ServiceLevelSpecApplicability defines when or how the SLA should be applied; for example, if there is an agreed outage, the SLA might not apply.

We shall now discuss how the SID might be applied by a DVB-T Service Provider in modelling the DVB-T Service Topology.

5.4.1.5 Resource Model of a DVB-T Network Topology

As a first step towards understanding the DVB-T Service Topology, one needs to understand how the service is constructed at the resource level.

The SID provides abstract Physical and Logical Resource models which are sufficiently rich to be extended to model in detail all of the equipment and signal paths in a network. However it should be noted that other network resource model standards exist which in some ways are more complete. For instance the DMTF CIM model (*DMTF 2006*) has been applied to concrete technologies and has open source model repository implementations (*DMTF 2004*). However the CIM does not provide a whole business view with linkages to the Service and Product layers. An inter-working strategy between these two models is being jointly addressed by the DMTF and the TM Forum (*TMForum 2005a*) (*TMForum 2005c*) and one would look carefully at using this if the intention was to model all of the physical equipment in a transmission station including power supplies, transmitters, receivers and the joining cables. However it is not necessary to go this far if one is simply considering modelling an SLA. Rather than try to model the complex signal path within each Broadcasting Station, one simply considers SLA's against the stations' input signals and output signals.

The lower right hand side of Figure 15 illustrates the SID linkage between Services and Resources. In particular one can see that two types of Logical Resource are of use; ProtocolServiceEntity which will be used to model the content of a Transport Stream (i.e. what is broadcast) and ManagedTransmissionEntity which is used to model the network topology (i.e. where it is broadcast).

5.4.1.6 Modelling a Transport Stream

The description of the DVB system given by (*Benoit, H. 1997*) has been used to construct Figure 16 which represents some of the complex relationships between the components of a multiplex which carries a DVB-T service. A DVB-T Multiplex is broadcast to a region on a specific RF channel. A 'bouquet' of several multiplexes may be broadcast to the same geographical region but the same program information (Network Information Table) covering all services in the region must be broadcast on all multiplexes in order for set top boxes to retune when they need to access a service in an adjacent multiplex.

Each Multiplex carries an MPEG-2 Transport Stream. Transport Streams are composed of multiple Packet Elemental Streams (PES), each being identified by a Packet Stream Identifier (PID). The PES streams carry Audio, Video or Private Data. The Program Clock Reference (PCR) streams carry common time codes to synchronise the Audio and Video streams. A Transport Stream also contains a series of Table Streams which carry tables to identify the content and relationships between the other streams in the multiplex. The most important table

is the Program Association Table (PAT) which identifies which programs are in the multiplex and how the set top box can access them.

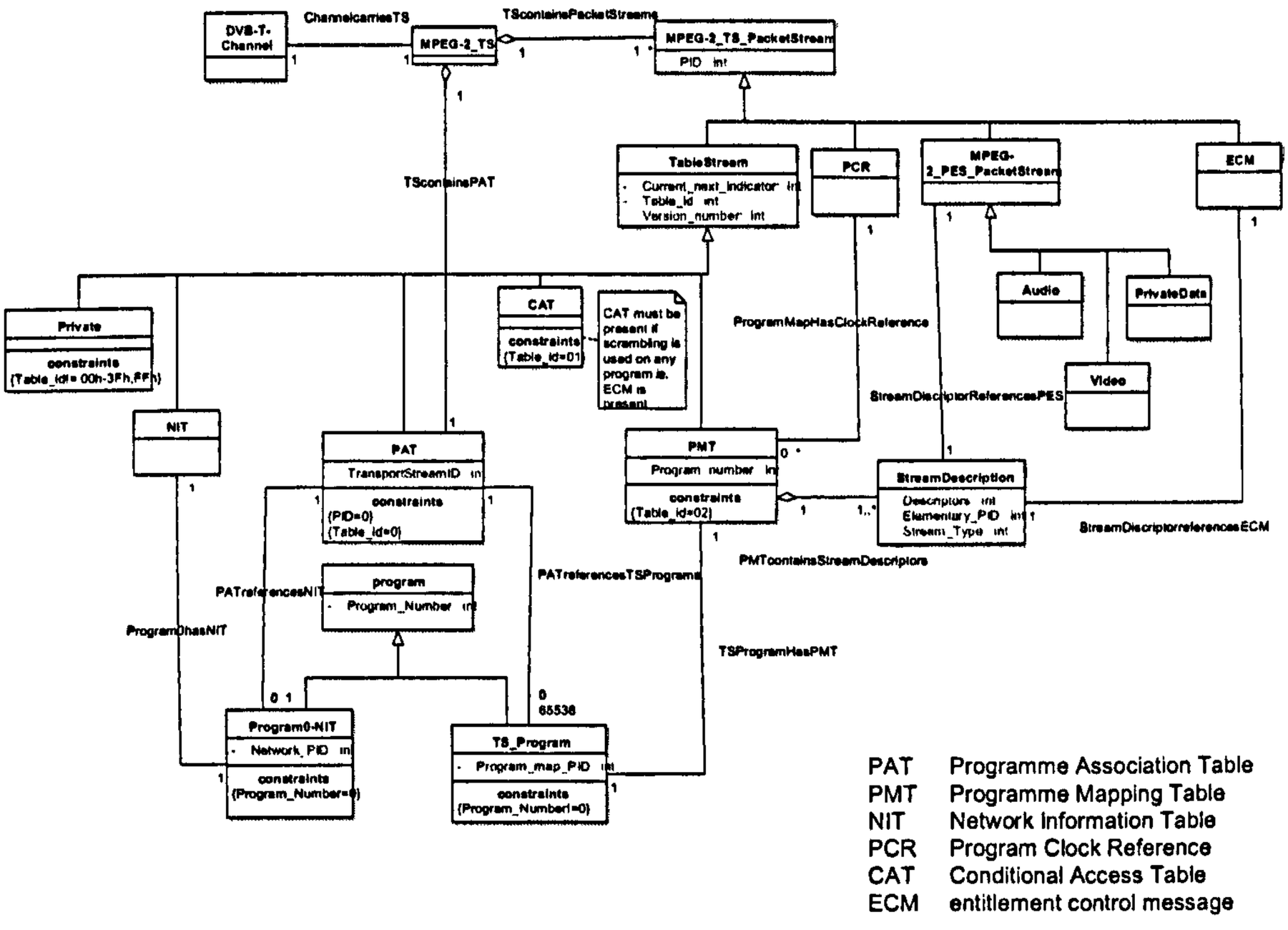


Figure 16 UML model of relationships between components of a DVB-T service

We need to discover an entity in the SID which gives us a starting point for defining the relationships between the components in a Transport Stream Resource.

The SID already defines Asynchronous Transfer Mode (ATM) as a WAN Protocol Service Entity and since Transport Streams are closely related to ATM (The MPEG TS packet size corresponds to eight ATM cells), Transport Streams might also be a candidate for modelling as a sub class of WAN protocol. However the SID defines WAN protocols as operating at the lowest three levels of the OSI model (i.e., physical, data link, and network) but the MPEG-2 Transport Stream is so called, to signify that it is the input to the Transport Layer (layer 4) in the OSI model. An MPEG-2 transport is a service multiplex. No mechanism, within the syntax, exists to ensure the reliable delivery of the transported data.

Thus in the absence of a more appropriate super class the TransportStream is modelled simply as an extension to ProtocolServiceEntity in Figure 15. The TransportStreamSpecification should then define the invariant relationships between tables and packet streams in a TransportStream and can represent a template for transport streams carrying certain content. A TransportStream instance would then model the actual content of a given transport stream in the network.

This is a useful construction since in practice Transport Stream Analysers (TSA's) are a type of test equipment located throughout the network which measure Transport Streams against their specifications as defined in the TSA's Transport Stream Templates. Transport Stream

Templates could be considered as a realisation of the TransportStreamSpecifcation. The resulting TS performance measurements are exposed through a standardised SNMP MIB (ETSI TS 102 032) (*ETSI 2007*).

Thus Transport Stream Analysers can generate Key Performance Indicators based upon the measurement of a Transport Stream against its TransportStreamSpecification. The service model can then contain Key Quality Indicators derived from Key Performance Indicators measured by each Transport Stream Analyser.

Unfortunately at present there appears to be no common standard adopted between the manufacturers of Transport Stream Analysers for defining Transport Steam Templates and so a common model of a Transport Stream Specification would have to be created if more than one Transport Steam Analyser type was used in the network (quite likely). Going forwards one could investigate whether the models underlying the JSR 927: Java TV API (*JCP 2006*) and Open-MHP (*Open-MHP 2007*) could be of use in modelling a generic Transport Stream Specification.

5.4.1.7 Modelling a DVB-T Network Topology

A Network is modelled by the SID as a Logical Resource representing a collection of Physical Ports which can be associated with equipment at geographical locations in the Physical Resource model. At the highest level of abstraction a DVB-T network can be characterised as a series of physical ports which also represent the Service Access Points (SAP's) for the purposes of SLA definition. The Broadcasting Stations in the network are modelled as Physical Resources with associated input and output ports. The input ports are aerials receiving the upstream off-air program (or ASI (*ETSI 2006c*) ports receiving a base band transport stream from the broadcaster). The output ports are transmission aerials.

In order to characterise the quality of service for the signal transmitted by each station, Resources representing notional viewers' home receivers are also included in the model. The quality of service in a region is thus tied to the service as perceived by the set top receiver object. Note that a key benefit of having an object which represents the viewer's set top box is that it can represent both measurements derived from associated objects or real measurements from an off air receiver where available.

Historically the SID was somewhat sketchy on the definition of the network resources. Most of the SID content in this area was originally based on Strassner's DEN-ng model (*Strassner, J. C. 2004b*) and concerned the management of IP networks. However recent work by the MTOSI team has considered how a transmission network can be modelled in the SID (*TMForum 2006d*). Although the MTOSI work initially concentrated upon using the SID to model SONET/SDH layer networks (as defined in ITU-T G.709 (*ITU-T 2003*)), the concepts can readily be extended to DVB-T broadcast networks. The fact that this modelling extends work already done for transmission networks is useful because the transmission network

between the studios and the transmitter network is likely to be SDH based. If it was necessary to consider inter-working between DVT and SDH management systems, one could leverage the layer mappings between DVB Transport Streams and SDH already defined by ETSI (*ETSI 1998b*)

Figure 17 shows a DVB-T network resource model as a UML class diagram constructed using SID Business Entities.

The SID has a concept of Administrative Domains which are used to characterise networks made up of sub-networks and layer networks. Administrative Domains are used to partition networks according to policies of ownership or control. Thus sub-networks, nodes or even ports within nodes may arbitrarily fall into one or more Administrative Domains. Sub networks are used to model logically related collections of ports. Layer networks are used to model multiplexing of services within different bandwidth bearers.

In Figure 17 a DVB-T Network is simplified to contain two layers. The radio frequency MUX layer network (the DVB-T_broadcastMUXConnectionNW) consists of sub-networks describing the regions (DVB-T_StationServedArea) immediately served by each station. The Transport Stream layer network (DVB-T_TransportStream_Trail_NW) consists of sub-networks describing the region (DVB-T_TransportStreamRegion) served by the same Transport Stream (i.e. where the same program content) is broadcast.

Each SID network layer represents a collection of possible Connections between nodes at a given bandwidth. Trails represent circuits or channels which exist end-to-end across a network layer and which are made up of one or more Connections (transmission links) supporting the trail. Multiplexing is modelled by a layer network of higher bandwidth trails providing connections for a layer network of lower bandwidth trails.

The SID transmission network model defines Pipes and their sub classes, Trails and Connections which are derived from the ITU-T G.805 standard (*ITU-T 2000a*). Every node in a network has one or more ResourcePorts. Every ResourcePort in a network is associated with one or more Connection Termination Points (CTPs). Put simply, a Connection represents a single hop at a given bandwidth between CTPs (i.e. nodes) in the network. A unidirectional Connection may have one source CTP and one or more sink CTPs. Thus the RF transmission of a DVT-T MUX from a single broadcasting station may be represented as a Connection associated to a source CTP (i.e. transmitter aerial) within the broadcasting station. The receivers of this DVT-T MUX each have their own sink CTPs (receiver aerials) also associated to the Connection. A 'Bouquet' of DVT-T MUX transmissions transmitted from the same station (on different frequencies) may be represented as a sub-network containing several DVT-T MUX Connections.

(Note that strictly speaking the SID recommends using Resource Specifications and Roles to extend Resources rather than sub-classing. This should be applied consistently if the objective

is to contribute extensions to the SID standard. However if the SID is used only as an analysis tool, it is acceptable to use sub-classing to simplify the analysis. The present analysis has created sub classes of Ports and Pipes to illustrate the modelling of a DVB-T service).

Figure 18 is an instance diagram illustrating how elements of the model in Figure 17 can be used to represent the service topology described in Figure 14. In the diagram Platinum stations in two programming regions inject Transport Streams into the network. The DVB-T MUX transmitted from each station is represented by a Connection. Any Receiver of this broadcast is represented by a Connection Termination Point associated with the MUX Connection. The Transport Streams in each region are represented by Transport Stream Trails. In a receiver which can decode the Transport Stream, a Transport Stream Trail Termination Point is always associated with a MUX Connection Termination Point. A Transport Stream Trail passes straight through a station if no multiplexing is done in a station. However a Gold station in Region 1 has a protection switch which can choose to re-broadcast programming from Region 2 if its parent station fails. This is modelled by changing the associations between Trail Termination Points within the station object.

Note that the layer network model could be extended to consider a layer network for each of the Elemental Packet Streams carrying program content. However since most nodes in the network do not change the content of the Transport Stream, the added complexity of modelling the individual Packet Streams as a layer network on a per-node basis is not really required. Instead a simplification is to define a new Transport Stream sub network at each location where the mix of the Transport Stream can be changed and associate all the nodes in that sub network with a separate Specification of the services in the Transport Stream.

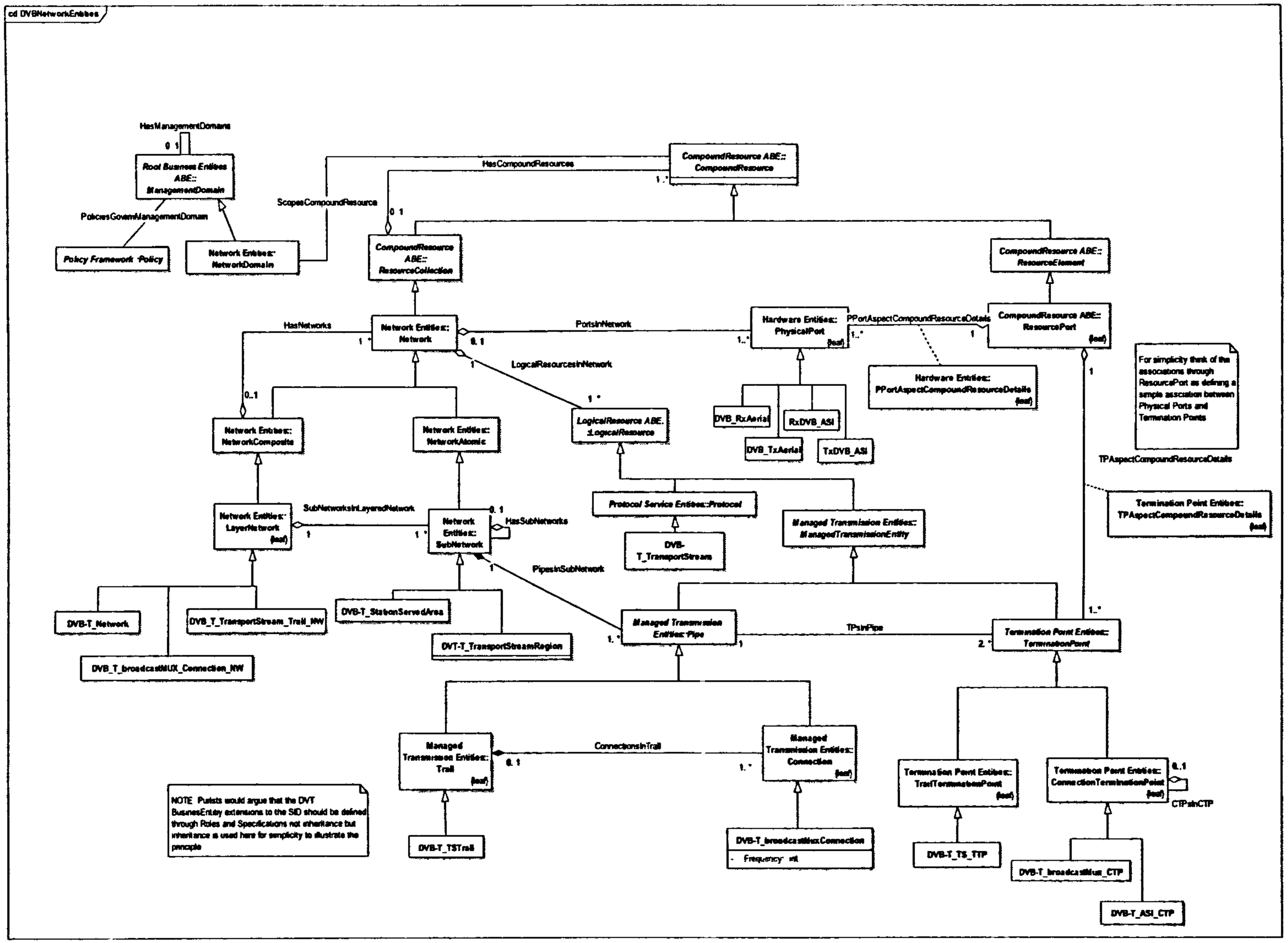


Figure 17 Modelling a DVB-T broadcast network in the SID

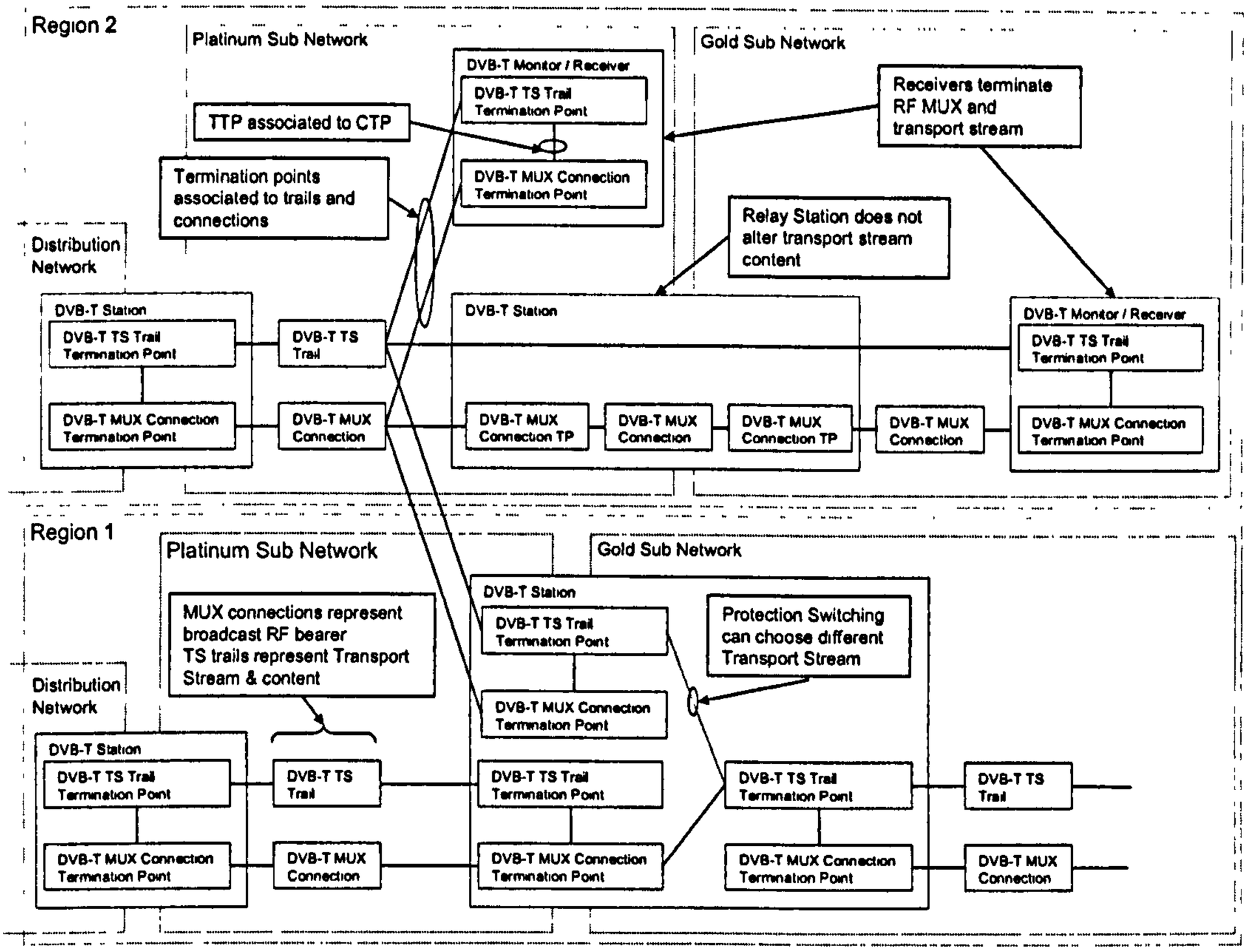


Figure 18 Illustrating how DVT-T MUX's and Transport Streams are modelled

In summary then, this analysis has shown that one can use the SID to derive a service model which includes the content and geographical distribution of transport streams across a DVB-T network.

It is important to realise that in this analysis the NGOSS SID is used as a tool to aid in a common understanding of the SLA problem. As the SID is an information model, it is not intended to be used for implementation. Rather it is intended to help develop a clearer understanding of the business problem before implementation concerns are brought in.

The process of trying to map the DVB-T SLA problem onto the SID information model has given the researcher and Arqiva access to the related Telecommunications Industry's understanding of similar problems and facilitated the introduction of key concepts found commonly in OSS systems but not often encountered in broadcasting.

We shall now turn to considering a NGOSS System View which begins to consider how the model might be implemented.

5.4.1.8 NGOSS System View of DVB-T Service Management

In order to make this analysis more tractable, the System View was created with the constraint that there were a limited number of COTS OSS packages available to implement a solution. Therefore rather than starting from a blank sheet, the process began by identifying from the Telecommunications Applications Map (TAM) (*TMForum 2006a*) the types of systems likely to be used to manage a DVB-T network. The current TAM (Release 2) is still quite immature

but provides a useful starting point for setting boundaries on the problem of selecting suitable OSS components.

The TAM describes over 50 types of OSS system from order capture to billing. However once it is built, a DVB-T platform should be relatively static when compared to the services in a telecoms network. Thus the key functions to be considered for automation in DVB-T SLA management are those supporting Service Assurance. In practice the TAM functional blocks are likely to be functions enclosed within larger product packages from different vendors. The exact mapping of functions will of course depend upon vendor choice but for the purpose of illustration 5 conceptual product packages are identified and described in Table 35.

Figure 19 illustrates the relationship between each package in terms of the TAM functional blocks envisaged for managing a DVB-T platform. The arrows indicate broadly the types of information flows envisaged between the applications for a DVB-T solution.

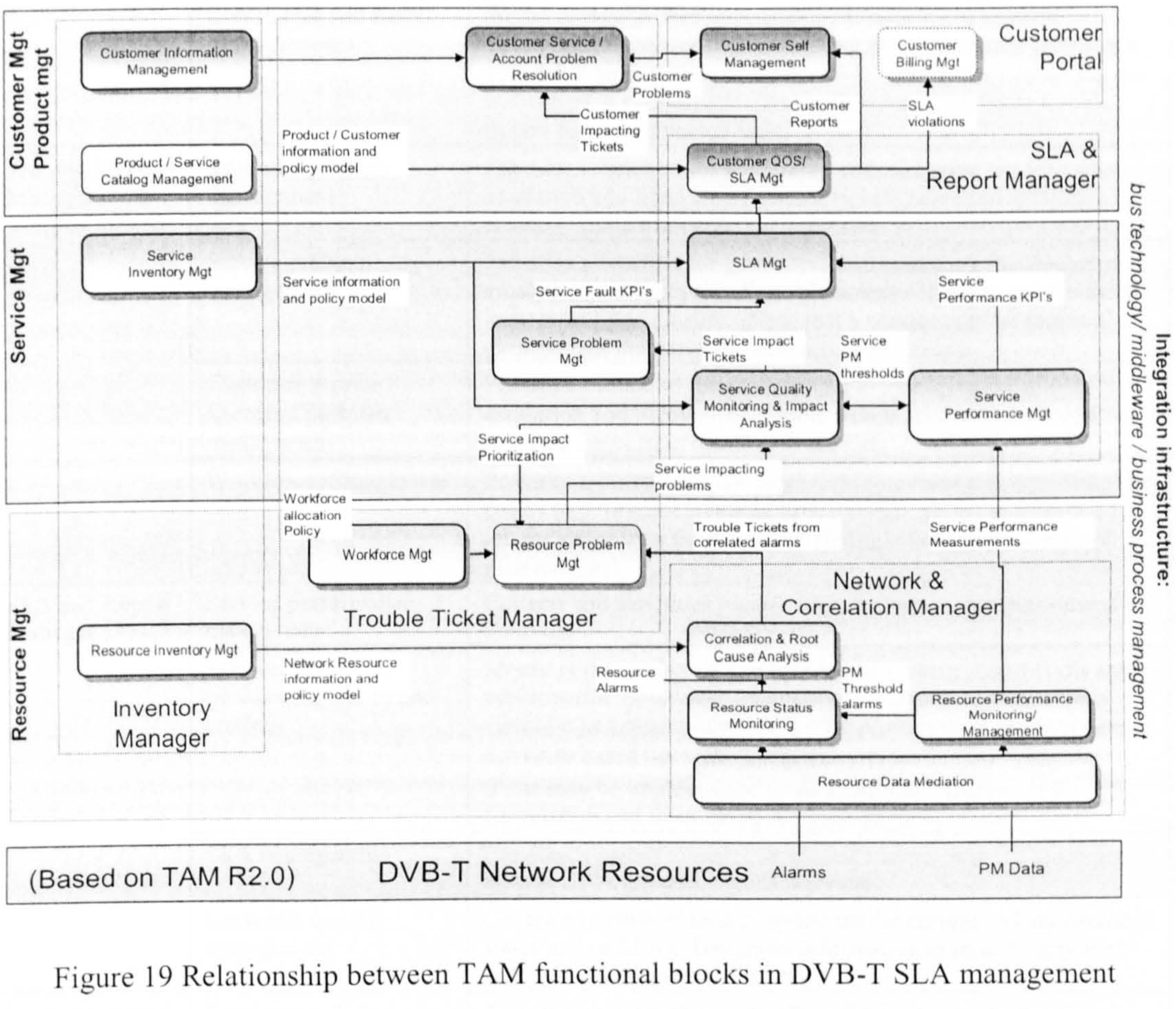


Figure 19 Relationship between TAM functional blocks in DVB-T SLA management

Table 35 DVB-T SLA management package descriptions		
Product Package	**TAM Function**	**Summarised Description of function**
Inventory Manager	Customer information management	Provides specific information about each customer; contact points, locations, escalation procedures, Products offered etc.
	Product / Service Catalogue management	Relates services to priced products offered to customers at a given price with a given SLA and SLA violation penalty policy.
	Service Inventory Management	Provides a service model relating network resources to offered services and SLA policies associated with services.
	Resource Inventory management	Provides a topology model of the relationships and dependencies between logical and physical resources used to create services and correlation rules (policy) for handling alarms.
Network and Correlation Manager	Resource data mediation	Maps vendor specific resource management protocols (e.g. SNMP MIB's) onto a common resource model.
	Resource performance monitoring / Management	Collects and processes performance management metrics from equipment. May generate performance threshold alarms according to a performance management policy.
	Resource status monitoring	Provides a common view and visualisation of the status of resources in the network. (In/Out of service / number of alarms etc.). Provides a manual means to generate trouble tickets related to a set of alarms.
	Correlation and Root cause Analysis	Works alongside Resource Status Monitoring to provide assistance to operators by correlating network alarms against the network topology and policy model to identify root cause alarms and suppress sympathetic alarms. Generates candidate trouble tickets for each detected fault.
Trouble Ticket Manager	Resource problem management	Provides a function for generating and managing the allocation, escalation and lifecycle of trouble tickets related to network resource faults from creation to closure.
	Service problem management	Provides a function for generating and managing the allocation, escalation and lifecycle of trouble tickets related to service faults from creation to closure. (Note that a service may be restored even though the network fault still needs fixed).
	Customer service /Account problem management	Provides a function for generating and managing the allocation, escalation and lifecycle of trouble tickets related to customer reported faults from creation to closure.
	Workforce management	Provides a function to allocate workforce resources according to a policy (e.g. present location, time on shift, skills) to a particular job indicated by a trouble ticket (fixing a fault or preventative maintenance).
SLA and Report Manager	Service performance management	Collects and generates reports of performance statistics related to a service
	Service quality Monitoring and Impact analysis	Monitors the quality of a service based upon reported faults and performance violations. Analyses the impact of a performance violation and recommends real-time prioritisation of restoration activities based upon SLA impact. (Note may be implemented as an additional function of the Correlation and Root cause Analysis engine).
	SLA management	Provides a means to map contractual commitments to customers against KPI's reported against services.
	Customer qos /sla management	Generates customer facing reports on the current and accumulated status and of SLA's. Generates information to modify customer billing to reflect status of SLA's.
Customer Portal	Customer self management	Allows customers to see specific information on the impact of network faults on their services and any other customer specific information.
	Customer billing management	Takes into account SLA violations when generating customer's bill. (This may not be an automated process).

The TM Forum are working towards defining standard NGOSS contracts describing interfaces exposed by TAM components but these are also in a very early stage of development. However as a short cut, two programs, OSS through Java (OSS/J) (*OSS/J 2007b*) and the Multi Technology Operations Systems Interface (MTOSI) (*TMForum 2005e*) have defined workable interface implementations which encapsulate key NGOSS concepts in interfaces which OSS vendors are implementing today. The interface specifications themselves are not full Contract Specifications for the underlying components but it is recognised that MTOSI and OSS/J are enablers for technology specific implementations of NGOSS standards (*TMForum 2004d*).

Thus we have shown how NGOSS can help in taking the functional architecture outlined in Table 35 and Figure 19 to the next level of design. The key to success is in developing an information model for the solution which relates concepts from the DVB-T network onto the required Service Level Agreements. This information model could be instantiated in a single Inventory Manager but it is not absolutely necessary to have single inventory repository provided there is a common understanding of how these systems are configured so that the information and in particular object naming is consistent across all of the systems.

5.5 Action Taking

In the Diagnosis and Action Planning phases, the problem of DVB-T SLA management was analysed using NGOSS principles. In the Action Taking phase it was necessary to consider how a trial could implement some of the concepts in this analysis and illustrate their use in a tangible manner.

5.5.1 Trial NGOSS Implementation View

5.5.1.1 Trial Scope

The previous sections have considered how NGOSS may be applied to modelling a DVB-T network. In order to further investigate how to manage a DVB-T network, Arqiva commissioned a proof of concept which sought to implement the key components of a DVB-T OSS solution using this approach. The digital TV proof of concept needed to demonstrate a significant reduction in the number of alarms seen by an operator and the mapping of these alarms into a service level model for the network.

The trial implementation illustrated in Figure 20 was constrained to address only the components contained in the Network and Correlation Manager described in Figure 19.

The Resource Data Mediation, Resource Performance Monitoring and Resource Status Monitoring functions were provided by OpenNMS (*OpenNMS 2007c*). Correlation and Route Cause Analysis was performed using a pre-production release of StateWise (*Sidonis 2007*), a very flexible alarm correlation engine, lent to the project by Sidonis. StateWise is designed to sit alongside any network manager, analyse the alarms against a network model using a set of rules and advise as to their impact. StateWise also implemented a basic Resource Inventory

and a rudimentary Service Quality Monitoring and Impact Analysis function. The trial added a basic integration infrastructure using OSS/J and JBossMQ JMS messaging.

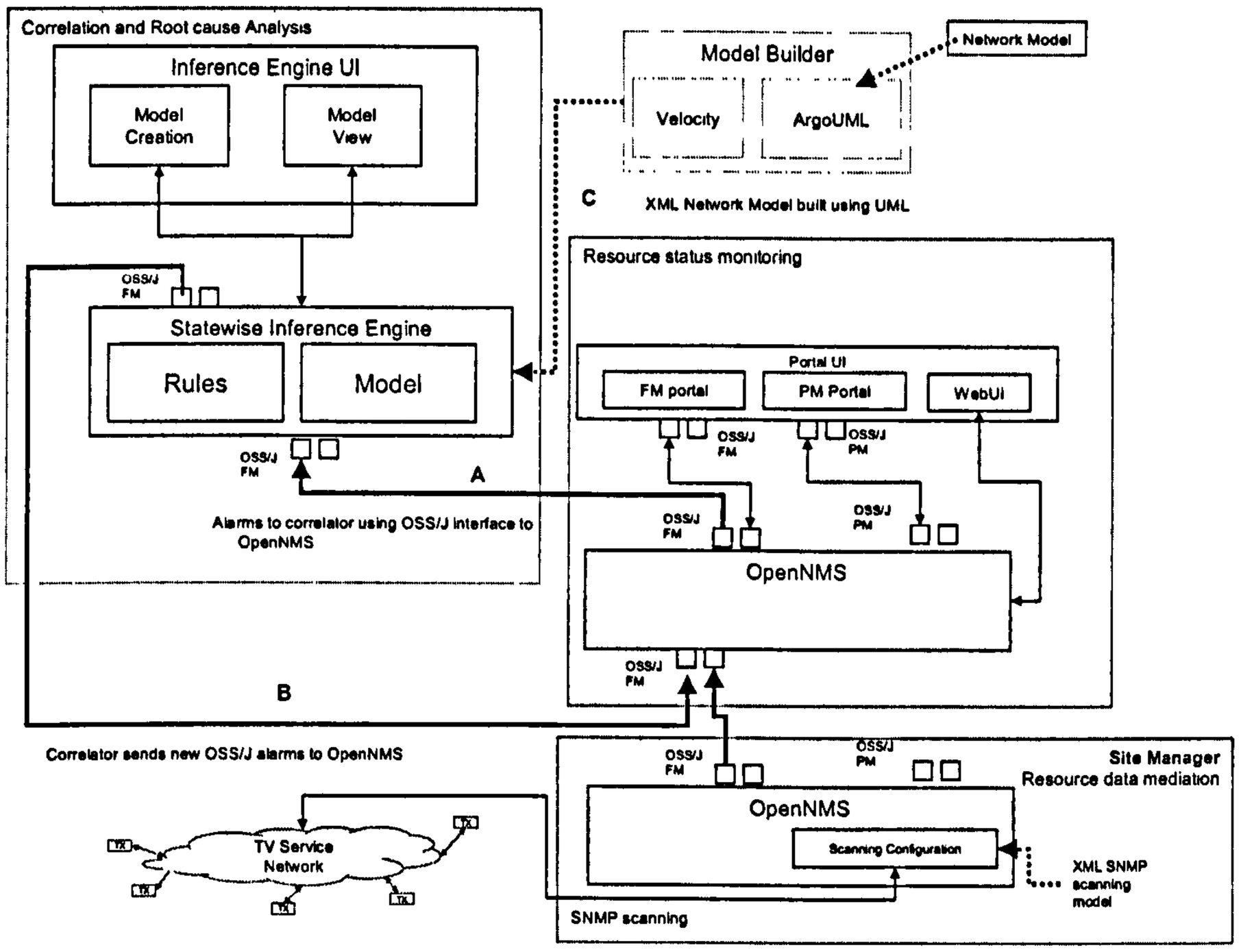


Figure 20 DVB-T Management Trial Implementation

It should first be noted that although StateWise is not an open source product, the designers were part of the trial team and so it was possible to modify and extend the product in the same way one could had done with an open source offering. Secondly, StateWise has itself been developed in Java using a large number of open source components. The designers, Don Kier and Martin Hobbs were quite clear that they could not have created the product without the free availability of the components they used.

StateWise uses rules in a correlation engine to determine the root cause and impact of the alarms given the network state. It uses an open source rules engine Drools (*Proctor, M., Neale, M., Frandsen, M., Griffith Jr., S., Tirelli Edson, & Meyer, F. 2007*) (now renamed JBoss Rules) to implement business logic. Drools is compliant to the JavaTM Rule Engine API JSR-94 (*JCP 2004c*). Originally StateWise used a commercial component called ILOG JRules (*ILOG 2007*). However Don Kier helped the Southampton University students during the OpenOSS Catalyst when they used the Drools correlation engine and as a result of that experience and meeting with the Drools designer Mark Proctor (*Proctor, M. et al. 2007*), Sidonis decided to switch to Drools for their product.

StateWise maintains an internal topology model of the network against which it maps received alarms. The topology model can be flexibly defined to map onto any arbitrary network technology. A UML defined meta-model is used to determine the structure of the topology model which can then be populated with instance information using an external database or

data in Comma Separated Variable (CSV) format (*Hobbs, M. & Kier, D. 2005a*). This flexibility allowed us to experiment with the type of model needed for a DVB-T network.

The model described in Section 5.4.1.7 was an NGOSS information model and not a realisable data model (*Pras, A. et al. 2003*). Part of the transition into the Implementation View was realising this as a Data Model suitable for the StateWise system. It was not possible, given the time constraints, to implement a full SLA information model but a basic correlation model was created onto which equipment alarms could be mapped to determine the service impact of particular faults. This is illustrated in Figure 21. Note that for StateWise, it was sufficient to include only the associations between key elements. The full inheritance hierarchy was not required.

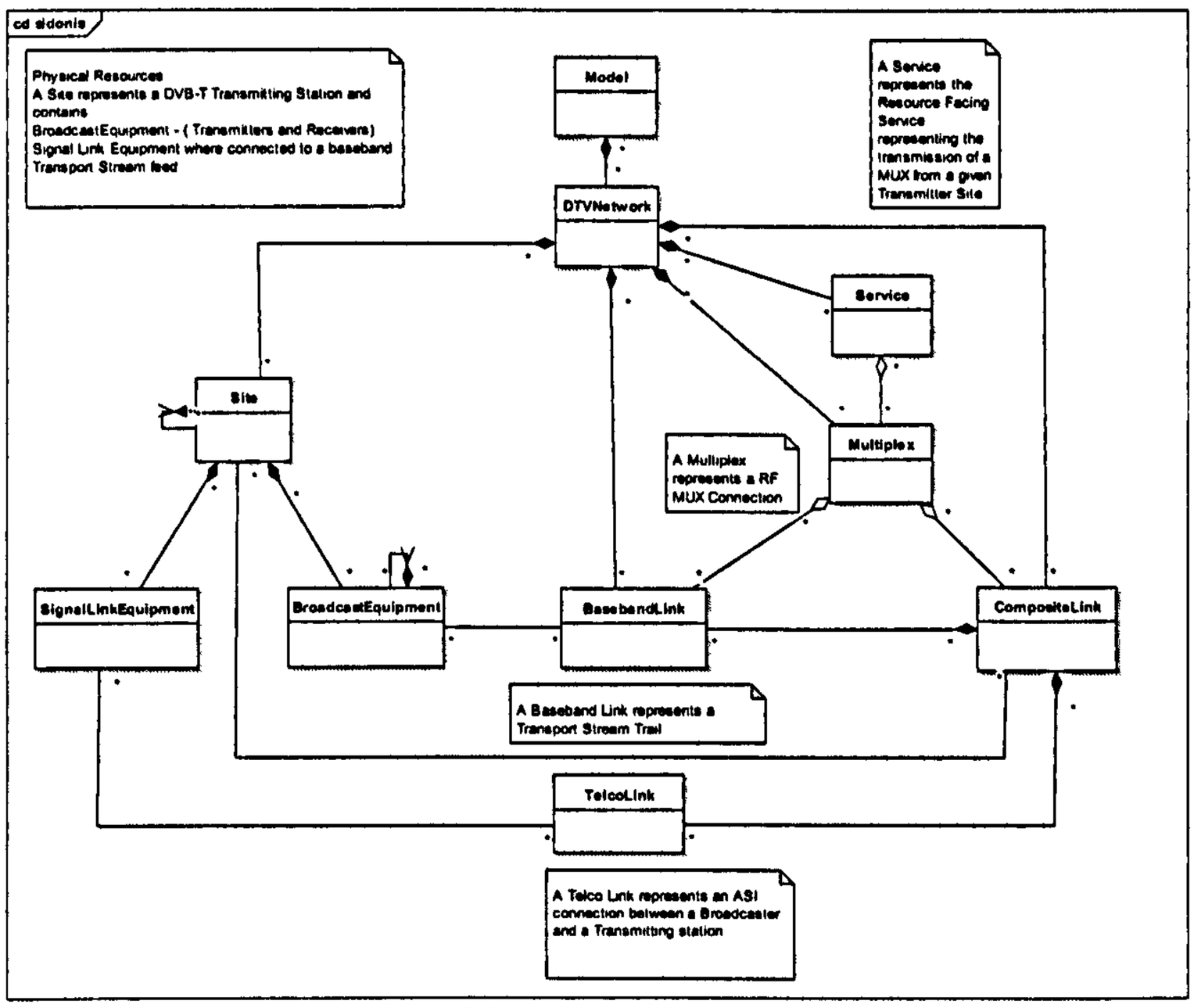


Figure 21 Data Model used in Proof of Concept
(adapted from StateWise User Guide (*Hobbs, M. & Kier, D. 2005b*)

Arqiva could have used an established COTS Network Management product to feed StateWise in their proof of concept (such as Micromuse Netcool (*IBM 2007b*), HP TeMIP (*HP 2007a*) or HP OpenView (*HP 2007b*)). However Arqiva wanted to use the opportunity to understand whether an Open Source alarm management framework which could be distributed around their 1000 unmanned sites would be feasible. Arqiva still intended to go out to tender for a full system but an Open Source demonstrator would provide a benchmark against which they could assess the benefits of COTS alternatives. The Open Source demonstrator would also be available for potential bidders to better understand the unique problem space in a DVB-T network. Thus the challenge was to create a prototype distributed alarm management framework which could monitor an SNMP transmitter network and feed StateWise with usable alarms.

The key design requirements for the Implementation View were as listed in Table 36. It is important to note that for the purposes of the proof of concept it was not required to demonstrate non-functional requirements such as scalability, software ruggedness and failover-standby capability. Furthermore it was only necessary to implement those parts of any standard interface (in this case OSS/J) which were needed for the trial. The trial did not require a fully conformant OSS/J solution.

Table 36 Requirements for Arqiva Proof of concept NGOSS Implementation View
1. A distributed architecture to consist of a 'central manager' and separately packaged 'site managers'. Each site manager would locally monitor the 100 or so diverse equipments at a transmitter station using SNMP and report exceptions as alarms to the central manager which would be connected to StateWise for Alarm Correlation and Service Impact Analysis.
2. Each site manager would have the capability of providing an overview of all equipment on site and provide a local management consol for Engineers. The consol would also allow rapid navigation to any web or CLI interfaces on transmission equipment. The Site manager UI would also be accessible remotely.
3. SNMP performance polling and trap management to be done on site near to the equipment. The de-duplication and mapping of traps into alarms to be performed on site.
4. Demonstrate the use of SNMP to manage a selection of broadcast equipment including DVB Transport Stream Analysers, DVB Receivers and Transmitters and local environmental alarms; Power, Fire, Door alarms etc.
5. A standard messaging protocol (not SNMP) was to be used over the wide area network between sites and the central manager. StateWise required alarms to be presented having fields conforming to the X733 (*ITU-T 1992c*) alarm definition. In particular it required each alarm to include fields for Managed Object Instance and Managed Object Class
6. Where possible the proof of concept was to be constructed using Open Source and freely available to any potential OSS bidders to Arqiva.

5.6 OSS/J Messaging Protocol

StateWise required an alarm definition conformant to ITU-T X.733 (*ITU-T 1992c*) and this capability was provided as standard by the OSS/J Quality of Service API (*OSS/J 2002b*). This was one of the first OSS/J specifications and is somewhat inconsistent with later specifications which are based upon a Core Business Entities model (*OSS/J 2006a*), (*OSS/J 2007a*) which is directly based on the SID. OSS/J is in the process of developing two new specifications for Fault and Performance. However these are still draft and it was felt that it would be better to target an existing specification and upgrade later if necessary.

OSS/J can be transported over the wide area as XML messages using any JMS provider. OSS/J also has a web services profile (*OSS/J 2006b*). However as this did not yet allow for the pro-active sending of events the project needed to use the JMS based profile.

JBossMQ (*JBoss 2007c*) was chosen to provide the JMS messaging service since this was a proven open source product. JBoss are currently working on a next generation distributed messaging architecture called JBoss Messaging (*JBoss 2007a*). This could be used to construct a distributed messaging infrastructure avoiding single points of failure in the future.

5.6.1.1 Site Manager

The Site Manager project provided the use cases guiding the contributions to OpenNMS which are covered in Chapter 6. Much of the basic Site Manager functionality was already provided by OpenNMS. It could monitor and display the status of local equipment on a web interface. Previously OpenNMS had simply managed SNMP traps as events, but in the previous 6

months they had introduced a very flexible event to fault mapping capability. Thus OpenNMS could be used to de-duplicate traps and map alarm raising and alarm clearing traps into OSS/J alarms corresponding to the current state of the equipment.

The alarm definition in OpenNMS did not fully match X733 and the researcher had to work with OpenNMS to introduce the Managed Object Instance and Managed object Type fields in order to be able to map the alarms onto the topology model in StateWise. It is important to note that this approach is completely model agnostic, it allows user configuration in OpenNMS to map SNMP derived alarms onto any chosen model be it proprietary, CIM or SID based.

One of the benefits of working with OpenNMS was that it was possible to leverage the fact that they were actively in the process of re-architecting their product using the Spring Framework (*Johnson, R. 2005*), (*Spring 2007b*). Spring provides a simplified object container with dependency injection which makes testing and configuration of an application using Spring much easier. Spring appears to have an assured life as it is being used by a number of vendors as the basis for their EJB3 implementations.

The OSS/J interface was written so that it could either run in a Spring or J2EE container. By running in a Spring container, the interface could run natively on OpenNMS and expose a JMS/XML based OSS/J interface directly off OpenNMS itself. Alternatively, the interface could run in JBoss (with an RMI interface into OpenOSS) and expose the JVT and JMS profiles of OSS/J.

This approach meant that OpenNMS systems running as site managers could run the interface natively and connect to a remote messaging provider (JBossMQ) which would be hosted with the Central Manager

5.6.1.2 Central Manager interface to StateWise

OpenNMS was provided with an OSS/J message receiver which could run multiple threads, each connecting to separate OSS/J JMS topic queues. This way a central manager based on OpenNMS could receive alarms from multiple remote OpenNMS systems using the same messaging bus.

The central manager provided an upwards interface to StateWise - again exposed using OSS/J JMS topics, which provided a consolidated view of the alarms for the whole network. StateWise was thus able to correlate alarms and create new 'service impact' alarms based upon its correlation.

5.7 Evaluating the Digital TV Proof of Concept

This chapter has demonstrated how the NGOSS standards could be applied to specifying a system for managing a DVB-T platform. The key functional components and the information model for a complete design have been outlined. A small scale mixed open source/COTS proof of concept has been used to implement some of the key concepts described. In

evaluating the exercise it is useful to consider the objectives which were jointly agreed and set out in Section 5.3.3.

5.7.1 Use of Standards

Arqiva wanted to gain practical experience of highly innovative OSS solutions to managing a DVB-T network while at the same time not prejudicing their later RFQ process.

In Section 5.4, the action planning phase of this project used NGOSS artefacts as tools to try and understand better the DVB-T SLA management problem. In particular the SID UML models were extended to characterise the DVB-T network and SLA relationships. Although ultimately there is a desire in the TMForum to use NGOSS UML models to expose NGOSS Contracts as blueprints and thus drive automatic code generation, the standard and the tooling are not yet capable of this task (see the discussion of NGOSS MDA in Appendix D.4.4). Instead this project has been an honest attempt to use parts of the current NGOSS standard as tools to help understand the DVB-T SLA management problem. The OSS/J interfaces were used as a shorthand for full NGOSS Contracts. This use of the SID has corresponded to Fowler's description of using UML as a Sketch (*Fowler, M. 2004*) where developers use the UML to help communicate some aspects of a system but not for code generation.

By tying the investigation to emerging standards, Arqiva were able to develop a framework of understanding against which they could later evaluate OSS product offerings and their own internal systems. However, had Arqiva tried to do this using just NGOSS or OSS/J alone, they would have found it extremely difficult to visualise the type of system to which their requirements were pointing. Therefore the provision of components which could be used to demonstrate aspects of the system's functionality was an essential element of the exercise. The use of standards showed that the solution was not tied to one implementation but could be replicated by any standards compliant system. Even if their prospective vendors had not yet targeted these standards, Arqiva could talk intelligently to them about the future trajectory of the industry.

Prior to the researcher's discussion with Arqiva, they were not aware as broadcasters of NGOSS, OSS/J or for that matter the TMForum as a body representing the OSS industry. Therefore the work has forged a practical contribution towards the convergence of telecoms and broadcasting.

5.7.2 Cost Containment

Arqiva needed to contain the cost of the trial and investigate low cost technology alternatives to a traditional OSS stack.

Arqiva could have decided to evaluate multiple OSS products outside of this exercise, but given the complexity of the OSS industry, they would not have had a framework for evaluation or the time to engage across multiple products. Furthermore, they would not have been able to get vendors to commit to integrated demonstrations of similar complexity to the DVB-Trial

without some level of commercial commitment. The use of open source allowed Arqiva to engage in a much deeper investigation of the problem than they could have under more conventional commercial constraints.

The trial did indeed show that open source could offer workable solutions at a significantly lower price than COTS equivalents. In the subsequent RFQ process, some vendors offered solutions at a price which was an order of magnitude greater than Arqiva's budget. The open source activity gave Arqiva the confidence to have robust discussions on pricing based on their understanding of the risks and complexity of the task. It enabled them to put a value on the level of risk which the vendors were taking on relative to Arqiva developing their own solution. The use of standards combined with open source implementations gave Arqiva the tools to exercise more control over the risks associated with the project as discussed in Section 3.3.

5.7.3 Dissemination of Information and Innovation

Thirdly, the engineers close to the SLA problem in Arqiva needed to educate their own organisation as to the problems of managing a large multi service network.

Prior to the trial, some parts of the organisation had assumed that managing a complex DVB-T network would merely be an extension of the work which had gone before. The trial exposed the true complexity of the problem and also provided a focal point for the technically interested parties from across the Projects and Operations departments of the organisation to discuss the requirements for a solution. This exercise informed the requirements for quality reporting needed in Transport Stream Analysers and the MIB requirements for transmission equipment.

The alarm correlation modelling work was the most challenging aspect of this project. Being able to demonstrate, discuss and refine a real instantiation of a model made a significant contribution to Arqiva's understanding of the problem. The trial laid down some basic principles for modelling but also led to the realisation that considerably more work will be required in order to create a fully operational DVB-T service model. Sidonis StateWise provided a model repository for the trial but this was tied to that product. The trial demonstrated the need for a separate model repository as an essential component of any future NGOSS demonstrator system. A model repository capability might have been provided by the NGOSSoil project which was initially progressing in parallel with the DVB-T trial but unfortunately never came to fruition (see Section 4.3).

Admittedly one component of the trial, Sidonis StateWise, was not open source. However the developers of the product were actively involved in the trial and it was for all intents and purposes as malleable as an open source project would be. Participating in a trial with close proximity to the end customer allowed them to gather more requirements for the product and also understand how their product needed to interact with the other systems. In addition,

Sidonis were able to pass back requirements to the open source Drools (JBoss Rules) inference engine project which they were using internally. Thus a direct technical engagement was facilitated using open source which would otherwise have been subject to the more restrictive marketing constraints as discussed in Section 3.6.1.

5.8 Related Work

5.8.1 NGOSS DVB-S Digital Movie Distribution

(*Mannaert, H. & Adriaenssens, P. 2006*) have reported on their work to develop the Cast4All Content Conductor, a Web-Based Operational Support System for Satellite Digital Movie Distribution using DVB-S (*ETSI 1998a*). The project was developed by researchers at the University of Antwerp under the sponsorship of the European Space Agency (*Mannaert, H., Adriaenssens, P., & Meyers, E. 2006*) and commercialised by an ISV called Cast4All[46]. It is relevant because it is broadcast media related, claims a degree of NGOSS compliance and was developed using the open source components listed in Table 37 (*Mannaert, H. & Ven, K. 2005*).

Table 37 Open source components used by Cast4all

Linux:	Operating system.
JOnAS:	the ObjectWeb J2EE application server.
Tomcat:	the Apache web server or servlet container.
Cocoon:	the Apache XML publishing framework.
Axis:	the Apache web services framework.
Ant:	the Apache build environment.
Swarmcast:	forward error correcting code from Onion Networks for reliable multicasting.
Pftp:	an application for real-time multicast transmissions with precise bandwidth control.
OpenSSL:	the underlying project for an enhanced transport level multicast security.
IPTables:	the preferred firewall on the servers.

The system uses DVB-S as the transport mechanism for secure content distribution of digital movies to cinemas for subsequent projection. Given the very large amounts of data being broadcast to multiple locations, the system provides mechanisms for scheduling the transmissions (*Mannaert, H. & Adriaenssens, P. 2005a*) and recovering from data loss due to perturbations in the DVB-S channel (*Mannaert, H. & Adriaenssens, P. 2004*). Synchronisation and management of the remote receiving stations is achieved using web services (*Mannaert, H. & Adriaenssens, P. 2005b*).

Unfortunately although (*Mannaert, H. et al. 2006*) refer to NGOSS and OSS/J, they do not show much evidence that they have actually used these standards actively in the design - and their other papers do not refer to these standards. This contrasts with the DVB-T trial where there was an explicit attempt to try and use NGOSS and OSS/J in the OSS design.

However they claim that the system can be regarded as a prototype for a more generic NGOSS due to the heterogeneous network infrastructure, the complex functional requirements and the fact that the OSS makes extensive use of modular software architectures and generic infrastructure software for web-based applications.

[46] Cast4All www.cast4all.com

As a system for managing time shifted delivery of content, the functionality differs significantly from that envisaged in the DVB-T trial. Firstly it is not concerned with real time SLA management, since the content can be re-transmitted unlike a live broadcast. Secondly, the transmission chain is a single hop (one satellite to multiple ground stations) and so down stream alarm correlation is not an issue. Thirdly, the Cast4All solution is as much a service delivery platform as it is a service management platform. One can argue that this simply illustrates the convergence of the technologies in the industry. Appendix D.4.1 notes that BT's distributed systems deployment architecture was using the same type of EAI middleware for customer management, service and resource management, and service execution (*Crane, P. 2005*).

However, although differing in a number of aspects, the work is important for several reasons. Firstly it illustrates that it is possible to build a commercial NGOSS-like system using open source component similar to those used in the DVB-T trail. Secondly, it indicates that others are thinking of applying NGOSS-like systems to managing a broadcast environment. Finally, (*Mannaert, H. et al. 2005*) offer some lessons in the commercial use of open source which will be contrasted with the experience of working with OpenNMS in Chapter 6.

5.8.2 Action research into information systems modelling

The present Action Research project has not taken a formal approach to analysing the NGOSS modelling process and lifecycle (*TMForum 2004a*). However (*Hoppenbrouwers, S. J. B. A., Proper, H. A., & van der Weilde, Th. P. 2005b*) reported in 2005 that they were planning to use Action Research to explore the general utility of modelling techniques.

The theoretical underpinning of their research is in communication theory as described in (*Hoppenbrouwers, S. J. B. A., Proper, H. A., & van der Weilde, Th. P. 2005a*) and in (*Proper, H. A., Verrijn-Stuart, A. A., & Hoppenbrouwers, S. J. B. A. 2005*) where they discuss its application to existing frameworks including among others, the Zachman framework (*Zachman, J. A. 1987*), (*Sowa, J. et al. 1992*) and RM-ODP (*ITU-T 1997c*) both of which form the basis for NGOSS.

Later work in 2006 has described a laboratory where they have explored a modelling process through interviews with practitioners (*Hoppenbrouwers, S. J. B. A., Lindeman, L., & Proper, H. A. 2006*). Their approach appears quite formal but does not appear to be tied directly into practitioner projects.

Hoppenbrouwers work is relevant because they are trying to understand the value or utility which practitioners derive from particular modelling techniques and languages. There appears to be an unchallenged belief in the NGOSS community that the direction they are taking in the use of UML and in particular that the use of the NGOSS lifecycle will lead to the most effective way for practitioners to plan future OSS systems. This is partly justified by the experiences of the FORM project discussed in Section 1.3.2. However a more formal Action

Research approach leveraging Hoppenbrouwers work may well provide interesting insights into how useful the NGOSS Lifecycle really is. This would be a useful area of study as NGOSS becomes more widely adopted by the OSS community.

Chapter 6 Intervention: Contributing an OSS/J Interface to OpenNMS

Chapter 6 Intervention: Contributing an OSS/J Interface to OpenNMS

6.1 Introduction

The OpenOSS Catalyst (Section 4.2) gave the participants some first hand experience of using open source to solve an OSS problem but it also clearly demonstrated that a comprehensive understanding of the benefits of open source in an OSS context would only be obtained through an in depth participation in a 'real' open source project. Since the NGOSSoil project (Section 4.3) did not reach an execution stage, the researcher needed to cultivate relationships with an alternative project which could address the same research questions.

Chapter 5 discussed the value delivered to Arqiva from using open source OSS components within a digital TV management proof of concept. The proof of concept enabled them to innovate in the design of their future OSS system. This chapter will look at the same problem from the perspective of contributing back to an open source project. The Arqiva proof of concept provided the use case which drove innovation within OpenNMS.

6.2 Diagnosis

6.2.1 Choosing To Engage With An Open Source Project

In this study, the researcher wanted to intentionally, from the outset, use the action research paradigm to examine the issues of practically participating in an OSS open source project. The majority of the literature concerning open source has tended to report the work of researchers external to the projects under examination, who have used tools such as developer surveys (*Lakani, K. R. & Wolf, R. G. 2005*), code analysis (*Ghosh, R. A. 2005b*) or mail list analysis (*Mockus, A., Fielding, R. T., & Herbsleb, J. D. 2005*), (*Kuk, G. 2006*). Although a number of practitioners have written about their experiences (e.g. (*Raymond, E. S. 1999*), (*DiBona, C., Ockman, S., & Stone, M. 1999*)), these have tended to be after the fact and again from a general point of view. In addition the economic analysis of open source has tended to look at the macro economic issues surrounding open source (*von Krogh, G. & von Hippel, E. 2006*) or the motivation of developers in large projects such as Linux and Apache (*Lerner, J. & Tirole, J. 2005*) rather than looking at the specific business issues surrounding the commercialisation of niche open source projects.

It was hoped that Action Research would allow the researcher to move beyond a general consideration of open source projects towards the particular issues raised by the application of open source in an OSS context. This approach was similar to that taken by the Software Engineering Institute when they investigated open source; conduct interviews, participate in open source development activities, workshops and conferences and study prior literature on the subject (*Weinstock, C. B. & Hissam, S. A. 2005*)

Using action research to engage heavily with a specific project had several implications. Firstly, the results would be obtained primarily through reflective learning about the experience, not by quantitative analysis of the technology itself. A particular open source project would obviously have a unique culture and dynamics but a project working within the network management space would be subject to the same external factors (market dynamics and technology constraints) as a commercial OSS product. By actively engaging with key members of the project in the context of making a solid technology contribution, it was hoped that general lessons would emerge about how to use open source in an OSS context.

Secondly, the contribution should be sufficiently complex that it represented a significant investment by the researcher both to learn the technology in the existing project and to subsequently make the contribution. In many ways the choice of project mirrored a commercial decision to adopt an open source project. A wrong decision would have significantly de-railed the research. The research would cover a number of stages; choosing which project to participate in, entering the project, learning the technology and the technology drivers, learning and adapting to the social context within the project, making a solid contribution and having the work accepted by the project.

The broad criteria for choosing which open source project to participate in are listed in Table 38. During the planning of the OpenOSS Catalyst a large number of open source projects had been looked at which could fulfil some aspects of an OSS system (Some of these are listed in (*Bruce, G. L. et al. 2005a*)). Many of the components evaluated were not in themselves OSS applications but rather software components which could be used to create OSS applications. For instance, the team knew of parallel evaluation work being done within Eurescom which was looking at how an open source workflow component (Jbpm) could be used with OSS/J within a provisioning scenario (*Silva, N., Ryan, A., Jorge, H., Reis, P. B., & Guerra, M. 2006*). However this work was not destined to become a supported product.

Other components such as SugarCRM could possibly be adapted to telecommunications usage (*Bruce, G. L. et al. 2006*) but Telecommunications OSS was not a core market for the application and would require a significant change in direction for the project. The researcher also investigated ComPiere, a J2EE based ERP application and was told that it had been used in Nigeria in a telecoms application[47] - but again this was not a core direction for the project.

In the end it was obvious that OpenNMS was the only project which met the majority of the criteria listed in Table 38. The key project members had practical experience of the telecoms OSS environment, and although they were not yet playing very strongly in telecoms, it was a strategic direction for the project. It was therefore decided to engage actively with the OpenOSS project with a view to making a significant contribution.

[47] Email discussion 10/8/2004 with Jorg Janke CTO of ComPiere, Inc. http://www.compiere.com/

Appendix E provides a comprehensive case study and technology description of the OpenNMS project. This chapter will describe how the OSS/J interface was contributed to OpenNMS

Table 38 Criteria For Choosing an Open Source OSS Project to Research	
Success Criteria	Description
• Project is sufficiently complex	A non trivial project is necessary to provide a realistic equivalent to a commercial OSS solution.
• Project is sufficiently simple to understand	The project must be understandable by the researcher within the time frame allowed by the research (2 years). This implies it is reasonably well documented and architected so that contributions can be made.
• Project is actively migrating to leading edge technologies	The project should ideally be actively leveraging the general open source innovation curve as represented by emerging component frameworks gaining acceptance in the wider open source community.
• Project has room for innovation	The project should still have problems which need solved - offering plenty of room for innovation.
• Project uses java	This was a practical requirement since; a) the researcher had attained reasonable competence in Java and didn't want to have to re-learn C++ or other languages b) OSS/J provided a means to demonstrate a level of NGOSS compliance and integration with OSS/J would be much easier if the native application was Java based.
• Project is established with a commercial support base	In order to gain insight into the real issues in creating a project to address the commercial needs of telecoms customers, the project needed to already have established a user base and a commercial support offering.
• Project will allow access to the community and the lead developers	Some open source projects have been reported to be quite elitist and difficult for a novice to enter without a long right of passage. The project needed to have an open culture which would encourage the researcher to become a welcome contributor
• An OSS use-case can be identified	In order to make a useful contribution to the project, it was necessary to be able to create a use-case for a feature which would satisfy the OSS research criterion and at the same time not move beyond the comfort zone of the project's stakeholders (i.e. a roadmap contribution and not a diversion).

6.3 Action Planning

The diagnosis section has introduced the OpenNMS project and summarised the reasons for using OpenNMS as a vehicle for researching the potential impact of open source in the OSS space.

The case study in Appendix E describes how after some difficult early years, OpenNMS has established itself as a viable project with commercial support and through its technology refresh program is positioning itself for further development. Furthermore, OpenNMS has demonstrated that an open source network management project can attract interest from a very wide community of both practitioners and researchers. The ideas encapsulated in OpenNMS can be seen to have driven innovation, even if the innovation has not always returned directly as contributions to the project. Furthermore, the users of OpenNMS have been enabled to innovate in the way they use the product and appear to actively consider how they can leverage in their own environments each new feature contributed to the project.

Although only beginning to be recognised as an OSS application within the telecoms industry, OpenNMS has attracted interest across the spectrum of the OSS market place and it has been adapted to solving some very complex network management problems in a cost effective manner. Therefore it appeared that OpenNMS was poised to make a greater impact in the OSS space if ways could be found make it more attractive and visible to that community.

Section 3.6.3 characterised a more comprehensive TMForum open source program as fulfilling a similar function to von Hippel's innovation toolkit and outlined a number of requirements for the resulting open source OSS components in Table 25. Most of these requirements were already being adequately serviced by OpenNMS's design, support organisation and evolution roadmap towards Spring. It therefore appeared to be a good starting point for beginning to create an NGOSS open source component.

User Interface • Single sign on – multi user • Tomcat Servelet, JSP • Proprietary MVC model • Un-published API to core	OSS Requirements • Integration with corporate desktop • Single sign on, security model, • User / Role / Managed Domain mapping • Industry standard MVC model / API

Inventory Management
• Resource
  • Node, Port, 'other information'
• Service
  • Port Protocols / Services
  • SNMP, HTTP, MySQL etc...
 	
OSS Requirements
• Service / Resource information model mapped to industry standards
• Richer model of hosted services
• Richer model for 'other information'
	Trouble Escalation Management
	• User and Email contacts
	• Notifications & escalations
	• Simple escalation logic not able to deal with complex service / customer / manager relationships
	
	OSS Requirements
	• User / Managed domain mapping
	• Trouble escalation policy on per service instance
	• Trouble management workflow / trouble ticket system integration

Resource & Service Discovery
• Resource
  • Discover ports using PING
  • Discover nodes using SNMP
  • Discover services by scanning
 	
OSS Requirements
• 'Discovery' API to align local model with external inventory
• Potential to integrate with a configuration management capability
	Service Assurance
	• Performance Management
	  • Node statistics
	  • Port statistics
	  • Thresholds
	• Fault Management
	  • SNMP Traps to Alarm
	
	OSS Requirements
	• Standardised API for PM and FM integration
	• Distributed polling solution
	• Per service polling & threshold policy can be altered 'on the fly'

Figure 22 OpenNMS Stand-Alone Features vs. OSS Requirements at Q4 2005

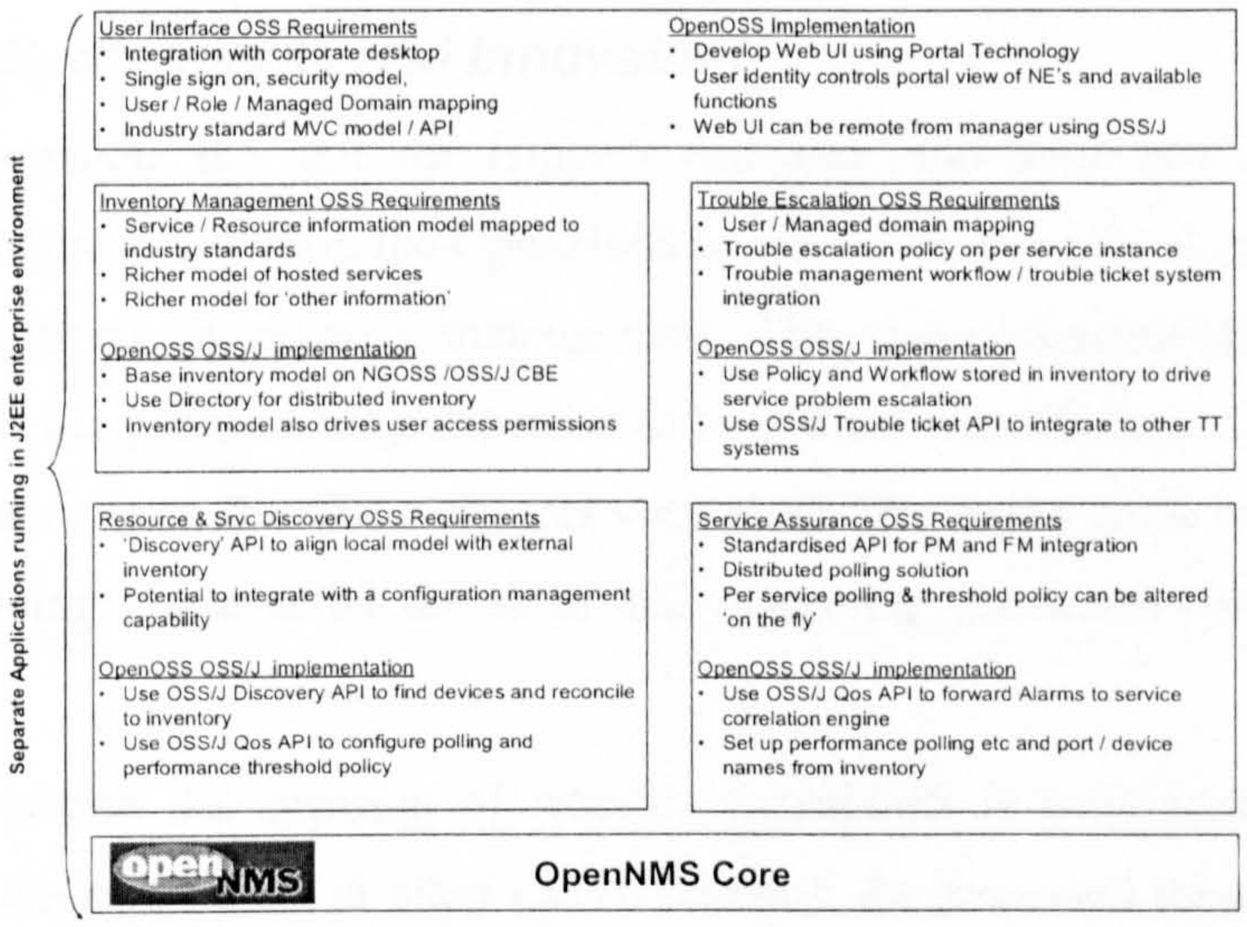


Figure 23 Proposed OpenNMS OSS Evolution

In October 2005 face to face discussions were held with the OpenNMS team to suggest ways in which the work already started to create an OSS/J interface could be taken forwards. OpenNMS was still being positioned as a stand-alone product which offered rudimentary OSS capabilities for Inventory, Trouble Management Service Assurance and Resource discovery (Figure 22). Could OSS standards be adopted to allow OpenNMS functionality to be

partitioned into separately deployable functional units, coupled using an OSS/J bus (Figure 23)? These ideas were not foreign to the project as in the long term the OpenNMS group wanted to develop separate Trouble Ticket and Inventory projects. However as a first step the researcher agreed to concentrate on developing an OSS/J Qos interface in order to demonstrate the power of the approach in a way which did not disrupt the overall project.

6.4 Action Taking

The action taking for this intervention was conducted in the context of the OpenOSS Catalyst and DVB-T projects already described. The intervention resulted in contributions to OpenNMS and the creation of a separate OSSbeans project (*OSSbeans 2007*). A detailed description of the design engagement with the OpenNMS project and the technical details of the OSS/J contribution are provided in Appendix F.

6.5 Evaluation

It is believed that this contribution to the OpenNMS project has created the first usable open source implementation of a full alarm management system using OSS/J. The project has demonstrated that it is possible to create such a system and that it can provide business benefit in a proof of concept. Having contributed the work as open source to OpenNMS, it is available for others to take forwards by implementing more of the OSS/J interface. This section will discuss the lessons which have been learnt from the exercise of becoming involved in OpenNMS and contributing the code. Lessons are also drawn from the OpenNMS Case Study in Appendix E.

6.5.1 OpenNMS, Standards and Innovation

It is clear from Appendix E.5 that von Hippel's end user innovation processes described in Section 3.6.3 are happening within the OpenNMS project. The majority of the contributors are end users or consultants in network management. The role of OpenNMS Architect, Matt Brozowski is to use his programming expertise to help these contributors turn their ideas into quality code. It is also clear that the end users very much like being close to the development process and not being hindered by the sales and marketing processes discussed in Section 3.6.1.

Appendix E.5 considers the question of whether OpenNMS is truly innovative or simply copying features already existing in other COTS systems. As discussed there is undoubtedly a degree of copying going on as OpenNMS catches up with other industry incumbents. However OpenNMS is being very innovative in its adoption and use of the work of other open source projects. Thus the innovation is partly in new features but also very much in aggressive cost reduction through the implementation of network management features using low cost technology. This innovation extends to pioneering the market for low cost OSS business

models (Appendix E.5.1) which match the risk profile of the customer as discussed in Section 3.3.

One needs only to look at the benefits gained through the DVB-T project outlined in Chapter 5 to see the potential value of having NGOSS-like components which enable end users to learn how to specify a system using NGOSS. As an enabler for NGOSS, the OSS/J standard is still in the early adopter stage and needs help to 'cross the chasm' to mainstream take-up as discussed in Section 3.3.1. It is hoped that the work done with OpenNMS will assist the innovation diffusion of the OSS/J standard by helping others who are evaluating OSS/J.

The DVB-T project is only one project and the OpenNMS OSS/J interface has not yet had time to experience wider adoption outside of this work. Wider use of the OSS/J interface will possibly take some time because firstly, the OSS/J standard has itself to experience wider adoption within the telecommunications industry. There are few if any other systems which could currently integrate with OpenNMS using OSS/J. Secondly, the current generation of users of OpenNMS are still very much wedded to using the event list rather than the new alarm list. This is partly because the OpenNMS notifications system is still tied in with events rather than alarms and most mainstream OpenNMS users integrate to other systems using this route. As interest in the alarms feature grows, so will interest in integrating with the alarm list.

This does not appear to be a problem for OpenNMS since previous projects like the Maps discussed in Appendix E.5.3 or JMX management discussed in Appendix E.5.5, have had a long gestation period. As discussed in Section 3.3.2, one of the benefits of the open source approach is that one can match the investment to market demand. Enough work has been done to prove a concept but the full implementation can progress more slowly until demand picks up. The work has created a Real Option in the code which has raised the value of the overall project for the OpenNMS community. This is evidenced by the fact that the OpenNMS group have recently been approached by several large vendors investigating OpenNMS who were in part attracted by the OSS/J work[48].

While OpenNMS's use of OSS/J may help the standard, working with the standard has certainly helped innovation in OpenNMS. Prior to the OSS/J Qos interface, innovation in OpenNMS was driven mainly from user requirements which were discussed in terms of extending the current product. OSS/J has widened the project's horizons by introducing the possibility of innovation also being driven by interaction with the standards community. This will help protect OpenNMS from the 'marketing myopia' (*Levitt, T. 1960*) caused by not looking outside the local project.

OpenNMS has now joined the TMForum which opens the possibility of OpenNMS extending the Qos interface to do Performance Management and investigating other OSS/J interfaces

[48] Private Interview with Tarus Balog at OpenNMS Dev-Jam August 2007 (on tape)

such as Inventory (*JCP 2007*) and Trouble Ticket (*OSS/J 2002a*) as the team begin to split OpenNMS functionality into separately deployable units as discussed in Section 6.3 and shown in Figure 23. This process could significantly increase the project's attractiveness to service providers and also begin to attract a wider developer base from within the standards community.

6.5.2 Reflective learning from adding OSS/J to OpenNMS

As discussed in Section 2.1.6, Action Research has enabled the researcher to reflectively learn from the experience of contributing to OpenNMS.

6.5.2.1 Initial Open Source Project Selection Process

Section 3.6.1 discussed the issues of using open source as a bridge between marketing and development. The researcher's overall experience of becoming involved in OpenNMS can be described as a journey from marketing towards development. The researcher was accustomed in his previous Product Management work to apprehending knowledge of the industry at a fairly superficial systems level; dealing in the features and benefits but not the technical details of a wide selection of OSS products. There is however a considerable difference between creating a systems design 'Marchitecture' (*Wikipedea 2007*) and actually getting it to work with open source components.

(*Giacomo, P. D. 2005*) has noted that finding an open source product can be more problematic than a COTS product because there is not usually a brand or marketing machine out seeking to attract customers. Tarus Balog[49] claims he no longer chases the OSS industry press because OpenNMS cannot compete with what he calls the 'canapé effect', whereby reputedly independent analysts are entertained and given stories by the major players. He considers it laziness on the part of the reporters when they do not research the evolving open source story and give it a fair hearing.

The onus is therefore on the potential adopters to find the project. (*Norris, J. S. & Kamp, P. H. 2004*) have also described the necessity to make many 'web shopping trips' to find the correct open source components. The researcher found that it was also necessary to make frequent return trips because the list of available open source components changed remarkably quickly. There is a need to have a background task continuously looking both for new components and unexpected new contributions to old components which can be leveraged in a present design.

One can envisage that, in the limit, as the open source model matures and more online information is made available about technologies, a component developer's problem will increasingly become one of choosing which technology to investigate and how much effort to invest before deciding whether to use it. Strategies are needed to guide the investigation. (*Edwards, S. L. & Bruce Christina 2002*) have described how the Action Research paradigm of

planning, acting, recording and reflecting can be applied as a model for effective internet searching. Although not approached as formally, this was very much the pattern which the researcher used when accessing information on open source projects. Since internet searching appears to be an important part of an open source developer's activities, it would merit further study to understand what is going on.

The researcher found that working in the world of open source required becoming comfortable in working with limited knowledge. One can only know so much about the many technologies available and one cannot know everything before making a choice. Once a decision is made to use a project, the users aim is never to have a comprehensive knowledge of the project but rather to learn just enough to get it working in the user's context. It is necessary therefore to have clear goals in terms of the problem needing to be solved and a strategy for searching the internet for other instances of the same problem. This would suggest that the potential adopter of an open source technology is primarily looking for similar use cases to their problem. The use case is the hook which attracts them to look at the technology in more detail. This implies that it is important for an open source project to document many user experiences and examples to show the potential adopters where the technology might be applied.

6.5.2.2 The Technical Evaluation Process

Evaluating an open source project requires installation and evaluation of the code on a local machine. In the case of OpenNMS, although it came with fairly comprehensive installation instructions (*OpenNMS 2007b*), these did not document everything about installation on a particular Linux distribution. The researcher was required to learn first how to install Java, Tomcat, PostgreSQL, RDDtool and a number of other dependencies on a Fedora Core system. In practice each of these components also had dependencies or configuration settings which took time to figure out. The researcher found that by learning through searching the project mail lists and looking for similar problems most of this issues could be overcome. However the first basic installation of the product required several days of work. To make a fair comparison, it should be noted that most COTS OSS components are similarly complex to install and the installation process is usually a professional service provided by the COTS vendor. In the case of OpenNMS, the user has the choice of paying for such services or learning for themselves.

The early experiences of developing for OpenNMS were equally frustrating. The process of setting up the development environment was considerably worse than an evaluation installation (although the later introduction of Maven 2 (Appendix E.4.3.2) has made it much easier to manage development dependencies). The early OpenNMS project had a poorly defined architecture and no documentation with the code. The web application was

[49] Private Interview with Tarus Balog at OpenNMS Dev-Jam August 2007 (on tape)

particularly complex and monolithic and the JDBC calls to the database were scattered throughout the server and web application code. One had to spend a considerable amount of time trawling through the code base in order to begin to understand how the application worked. One would observe however that many open source developers don't seem to find this a problem and actually welcome the challenge. The other early contributors to OpenNMS must have had the same difficulties but made a similar investment to become knowledgeable in the code before making contributions.

The NGOSSoil team were very concerned to release Nortel's code with COTS component quality documentation to in order to attract developer interest. This was a major cost which Nortel felt they would have to undertake in their project plan (Section 4.3.3). However the experience of the researcher suggests that open source developers have a much higher pain threshold than one might expect and are willing to live with poor documentation provided the code is available for analysis.

It is worth finally noting that the corporate members of the OpenOSS project and Arqiva all had difficulties in doing open source evaluations because of 'one size fits all' corporate IT security policies. Source code could not be downloaded and executed or contributed back to remote sites. In Section 3.5.2.2, Table 19 listed Security as one risk area where *(Li, J. et al. 2007)* found COTS and open source differed. Although (*Anderson, R. 2005*) has suggested that there are no inherent advantages of open source or COTS projects in terms of security, the experience of this project suggests that the issues lie more in the infrastructure and processes which the open source adopter applies in their organisation. Thus the security implications of open source OSS adoption policy in telecommunications would be worth further study.

6.5.2.3 The Value of Introducing Component Frameworks

As discussed in Appendix E.5.2, OpenNMS has introduced a modular architecture in expectation of a growing developer base. (*Johnson, J. P. 2002*) has created a model which suggests that a small developer base can continue to function fairly efficiently with a non modular architecture however a large developer base needs a modular architecture in order to be able to contribute more code. In a non-modular environment, if a developer wants to do something they must be willing to make all of the changes needed across the code base. This is the mode in which OpenNMS had to operate prior to the introduction of Spring (Appendix E.4.1.5).

As a contributor, the researcher found that the transition which OpenNMS made to Spring, Maven and Subversion was painful because each change required a major re-factoring of the contributed code and delivered no immediate value in terms of functionality enhancements. Conversely the move from CVS to Subversion allowed much finer control over the version of the OpenNMS code base which was being used for integration - making it possible to see

changes on a per-commit basis and work out which changes had caused the OpenOSS code to fail.

Maven and Spring have also made dramatic improvements to the readability of the OpenNMS code and the ability to contribute enhancements. Although OpenNMS have made some efforts to improve their own documentation, the adoption of Spring and Maven has also tangentially resulted in a great improvement in OpenNMS's development documentation because of the pre-existing books and web sites which describe how automated build and testing is done in Maven and how Spring integrates otherwise disparate components into a comprehensive framework. Thus the adoption of these frameworks has made it much easier for a new developer to get involved in OpenNMS. One lesson which can be learnt from this is that an application like OpenNMS is, above all, a consolidator of other open source technologies. Although an application (COTS or open source) may still offer unique user functionality, the design choice to leverage an underlying widely adopted open source framework can dramatically increase the number of developers who could begin contributing to or integrating with the project.

One would also observe that the value Spring adds is in simplifying the choice and integration of sub components. There appears to be a symbiotic relationship between Spring and its sub components in that each sub component sees the value in being part of Spring and the Spring project standardises their use in a larger framework. Because of the wider usage the components enjoy (a network effect), it is likely that they will be better maintained and will take note of changes which might otherwise break their integration into Spring.

The relationship between the open source component frameworks such as those used in OpenNMS and the underlying components projects is illustrated in Figure 24. One can observe a pattern emerging where the introduction of the Java platform lead to the creation of numerous stand alone open source component projects which applications like OpenNMS had to integrate themselves. When applications like OpenNMS needed to create their own components - they often spun them off as separate projects (e.g. the SNMP library (*JoeSNMP 2007*)).

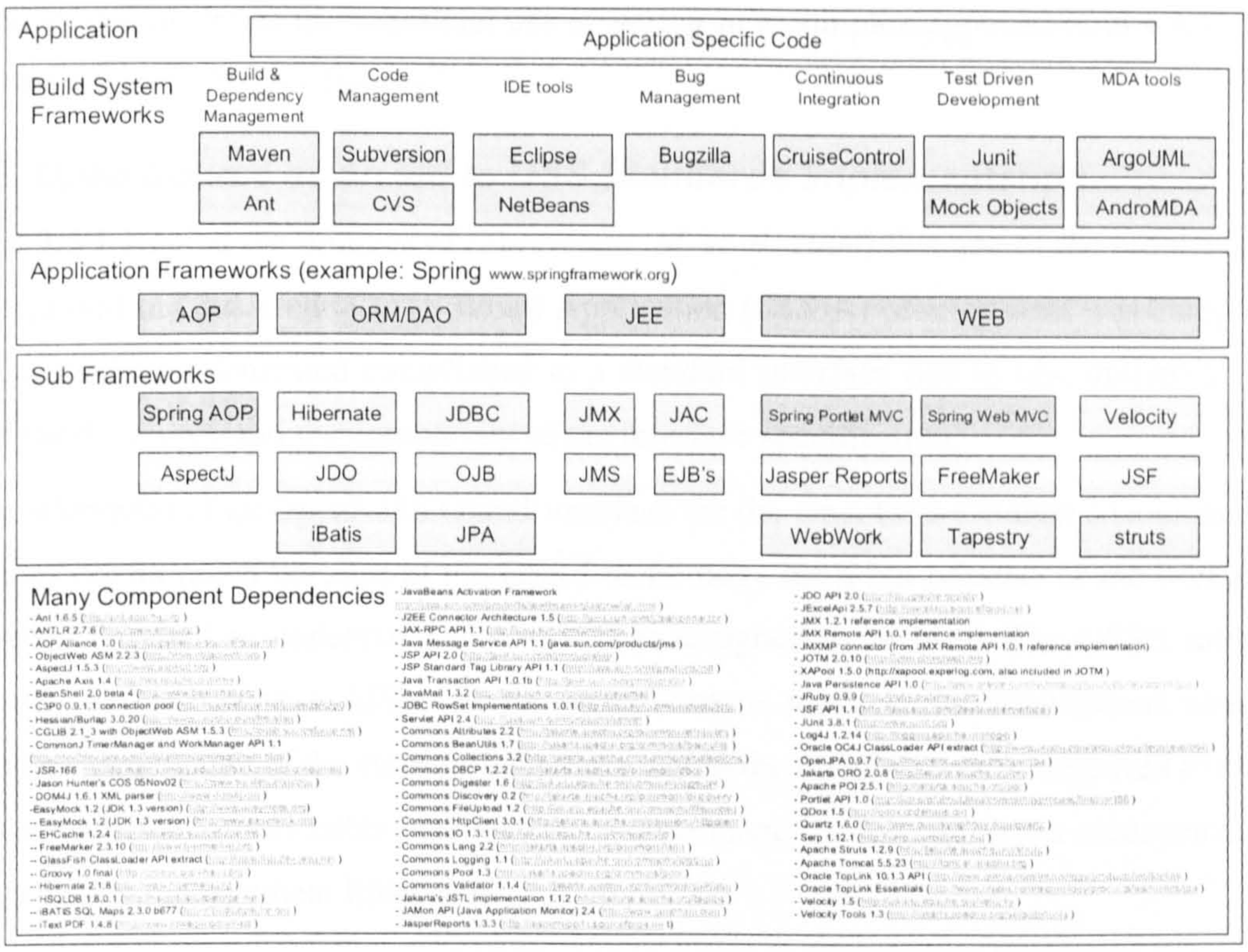


Figure 24 Illustrating Dependencies between Applications, Frameworks and Component Projects

As the open source market developed, frameworks began to emerge which leveraged existing projects and solved common problems in a consistent manner (e.g. Hibernate solved the problem of Object Relational Mapping). Later application frameworks like Spring have integrated these sub frameworks such that increasingly there is less of a requirement for application developers to write enabling code not directly related to their application. In addition as (*Norris, J. S. et al. 2004*) points out, the underlying frameworks are written by developers with expertise in that specific technology - allowing the application developer to concentrate on the problem to be solved and not esoteric aspects of cache management or real time programming.

It is important to note that Middleware vendors have also seen the benefit of leveraging component frameworks such as Spring. BEA have included Spring within their J2EE WebLogic application server because of the improvements it makes to the development experience (*Piper, A., Hsiao, E., Johndon, R., & Wall, C. 2007*).

OpenNMS's move to adopt Spring, Maven, GWT and other significant technologies has also given them a very good starting point if they move forwards with developing the Inventory and Trouble Ticket applications discussed in Section 6.3. In evolving OpenNMS they have, in effect, been integrating the components of an open source OSS Product Line Architecture (*Greenfield, J., Short, K., Cook, S., Kent, S., & Crupti, J. 2004 p 376 ff*) which they could use to address other functions in the OSS problem space.

Finally, it is worth noting that the synergy with OpenNMS is also of benefit to the Spring Project who are now considering using OpenNMS as reference project for training purposes.

OpenNMS demonstrates the beneficial use of Spring in a complex application in a way which cannot be done by COTS adopters[50].

6.5.3 Open Source as an aid to OSS Standards Implementation

Section 3.5.1.1 noted the findings of (*Torchiano, M. et al. 2004*) that one of the problems most often reported in Glue-Code COTS Based Application (GCBA) development was components which wrongly implemented compliance to a standard interface due to un-familiarity on the part of the developer and the complexity of the interface specification.

The development of an OpenNMS OSS/J interface for the OpenOSS Catalyst encountered this problem - not so much because of the OSS/J technology itself but because of the background knowledge required to understand how OSS/J is designed to be used. The OSS/J interfaces were well defined as java API's but the specifications made many assumptions about the developers' knowledge of the systems behaviour which the interfaces were supposed to expose. This was particularly noticeable when trying to understand the overall Alarm management and Trouble Ticket management lifecycles.

Appendix F.1 described how initially OpenNMS did not have the functionality required to properly implement an Alarm list. The subsequent requirements for the Alarm list functionality were heavily influenced by discussing with Dave Hustace the issues raised when trying to create an OSS/J interface using existing OpenNMS functionality. This represents a very good example of the process of innovation discussed in Section 3.6.3 which (*von Hippel, E. et al. 1995*) referred to as a 'Templating process'. Here two very different and highly complex patterns; the OSS/J interface specification and the implementation of OpenNMS were brought together in the context of the OpenOSS catalyst project and the examination of the interference between these objects led to innovative improvements.

One would observe that any other developer trying to implement similar OSS/J functionality on top of their own OSS products will probably face the same problem. This can be explained by noting that the OSS/J standards have been developed by a small community with access to a limited selection of their own OSS products. Although the Prosspero program (*TMForum 2007g*) provides OSS/J test harnesses - these only test the mechanics of the interface but not the systems behaviour. It was noted when reviewing other standardisation efforts in Section 1.4 that the best specification in the world cannot make up for actually being able to observe the behaviour of a real system. As (*West, J. 2004*) notes, an implementer will face an extended trial-and-error process in order to discover how other firms have resolved specification ambiguities.

In dramatic contrast to this situation, once the OSS/J interfaces had been developed for OpenNMS it was possible to easily demonstrate their value and explain their use to Arqiva,

---

[50] Private Discussion with Dave Hustace, OpenNMS, London, October 2007

Sidonis and the OpenNMS community - who were not previously aware of or interested in OSS/J. Of course, the present OpenNMS implementation of OSS/J is no more normative than any other implementation but at least the particular interpretation of the specification embodied in OpenNMS has been made to work between two systems and demonstrates useful functionality. It can be downloaded and examined by anybody and the core OSSbeans code is available under a commercial friendly licence for anyone to use.

Thus it has been demonstrated that there would be real value in having functioning open source components which exhibit NGOSS behaviour rather than just stubbed reference implementations of an interface.

6.5.4 An antidote to Fear Uncertainty and Doubt

In Section 3.3 it was suggested that the marketing stance of many ISV's towards open source was to create Fear Uncertainty and Doubt (FUD). However this work with OpenNMS has lead the researcher to characterise effective open source activities as being motivated by the reciprocal of FUD; Courage, Conviction and Faith. What does this mean?

6.5.4.1 Courage

Courage in this context is defined as the willingness to confront fear, uncertainty and intimidation.

The classic example of this would be Stallman's willingness to risk his livelihood to start GNU (*Stallman, R. 1999*). In the network management space one can cite Tarus Balog's willingness to risk founding a business supporting OpenNMS customers after it was abandoned by its original sponsors or OpenNMS's users' willingness to risk sponsoring the project into their organisations.

The closed nature of the OSS industry (discussed further in Section 6.5.4.3) raises the barriers to entry from open source projects because it is easy for OSS incumbents to claim outsiders don't know what they are talking about. Intimidation is a natural consequence of limited knowledge. Section 2.2.2 noted that the decision to launch the OpenOSS project albeit with 'bounded rationality' also required a degree of courage (or foolhardiness) in the face of potentially being ridiculed by OSS incumbents in the TMForum.

The researcher has found that 'feeling dumb' is a common experience when working with open source. Most of the open source projects he investigated were initially inscrutable; representing huge complexity both in the code and in the development goals - which were often not explicitly stated and had to be discerned from searching each projects documentation and mailing lists before being able to ask the projects' members well framed questions.

Ultimately the decision to choose to use a project required the willingness to risk failure. For instance several weeks were spent experimenting with various open source XML parsing frameworks before a solution emerged for the OSS/J interface. During this time it was unclear

if the researcher was wasting his time. While good documentation practices can help, the reality is that any researcher getting involved in an open source project requires tenacity and the belief that understanding lies at the other side of complexity[51]

(*Beck, K. & Andres, C. 2005*) has described Courage as one of the 5 key values of Extreme Programming; Courage to persist with a problem until a solution emerges, Courage to throw away obsolete code despite the sunk effort and Courage to re-factor (i.e. design code initially to meet the immediate problem and re-factor later to improve the design). All three of these definitions of courage have been observed in open source practice.

The decision to refractor or significantly change a code base requires both courage and conviction. OpenNMS's decision to move to Spring and introduce a number of other technologies was a brave step which COTS organisations with an inventory of old code would be less likely to make. A company with a significant but old code base and a bounded set of developers has little incentive to refractor. Better to treat the project as a cash cow. (Thus the major re-factoring required before moving towards NGOSS is a risk which COTS vendors will naturally avoid). OpenNMS made their changes based on the conviction that a more modular code base would drive increasing innovation in the project by enabling many more (as yet unknown) open source developers to get involved.

Similar behaviour was observed many of OpenNMS's contributors. Although they would not always describe themselves as programmers, they were willing to take the risk to try and implement functionality. The culture in the project is to encourage and accept good efforts - and clean them up later if necessary.

OpenNMS's approach to engaging customers who wished to pay for development was to help them understand the workings of the system to the extent that they could appreciate the type of work that was required to implement their requested feature. This direct interaction with the development process compared favourably with the Product Consultant approach often used by OSS vendors outlined in Section 3.6.3. Rather than simply writing paper requirements specifications, ideas could be rapidly prototyped by OpenNMS consultants at customers' sites and later turned into statements of work. Thus the basis for development estimates were clearly understood by the customers and the customers also took some ownership of the risk of poor estimation and managed the consequences back into their organisations. This also avoided the customer being charged a risk premium for unclear requirements (*Chapman, C. B. et al. 2003 p271 ff*).

---

[51] To quote Oliver Wendell Holmes, Jr; 'I would not give a fig for the simplicity this side of complexity, but I would give my life for the simplicity on the other side of complexity'. http://en.wikiquote.org/wiki/Oliver_Wendell_Holmes,_Jr.

6.5.4.2 Conviction

Conviction in this context is defined as a strong persuasion or belief in the open source process. Whereas courage might be enough to make a move to open source, conviction is needed to stay the course.

Studies by, among others (*Lakani, K. R. et al. 2005*) and (*Ghosh, R. A. 2005b*), have observed that while some open source contributors do so out of a fundamental political conviction that open source is morally right (e.g. (*Stallman, R. 1999*)), there are actually a variety of reasons why individuals contribute. Similarly a survey by (*Bonaccorsi, A., Giannangeli, S., & Rossi, C. 2006*) has shown that the large majority of firms offering open source follow a hybrid business model, mixing products, types of licenses, and sources of revenues and that the positive experience gained by firms in working on community projects is much more important than ideological motivations of individuals.

The case study in Appendix E.1 observes that the early OpenNMS project exhibited a degree of hype and an expectation of short term financial returns which were not realised. However following the period of disillusionment (Figure 33), the project slowly began go grow based on the fact that it was generating just enough profit and user interest to sustain development.

Other developers in the OpenNMS project talk of Tarus as being a 'true believer' in open source and Tarus himself states that there is a significant difference between 'talking the talk' and 'walking the walk' in running an open source business. Tarus is quite critical of businesses which he observes claiming to be open source as a marketing stance but who are actually using the COTS model. Tarus would include in this definition firms which release a free version of their product with no support and an 'enhanced' commercial version. It is important to note however that Tarus also often says 'I am not a communist - I'm in this to make money'. His conviction is about the ability of the open source methodology to better serve his customers rather than an egalitarian desire to give it all away.

Several lessons can be drawn from this. Firstly, the observation that an open source project must itself be prepared to survive Moore's chasm between the early and late adopters discussed in Section 3.3.1. This can only be achieved if the project's sponsors have genuine reasons for maintaining the project through this period. In the case of OpenNMS, it was generating enough income for Tarus to survive in business - even if it was not generating enough revenue for the original OcuLAN business plan.

However the reasons need not be financial. Google have released the GWT project and BEA have released the XMLbeans[52] project because they needed the technology internally. Making them open source projects potentially attracted more developers but if not, the projects were still valuable to the businesses of their sponsors. Similarly OpenNMS has spun off JoeSNMP

as a project with wider applicability than just to OpenNMS. This same motivation lay behind the researchers design philosophy which separated the OSSbeans project from the core OpenNMS code as described in Appendix F.2.2. In all of these cases the sponsors already had a long term commitment to using the technology but believed it would become a better technology if others were using it too.

Thus conviction comes down to a firm belief that open source is strategically right for the particular business and must be married with the willingness (and ability) to stay the course with a project while it goes through the 'Trough of Disillusionment' (Figure 33) before it reaches critical mass.

6.5.4.3 Faith

Faith (or trust) in this context is firstly defined as confidence in the developer team and in the open source project and secondly a stance of acting in good faith towards the project. Confidence is primarily gained by participation.

The COTS model depends fundamentally on a contractually specified relationship between the vendor and the customer. Managers have historically preferred this model because they think it helps them manage the project risk. However the concealment of the code and the invisibility of the developers behind the marketing organisation ensure that there is always an imbalance in knowledge between the vendor and the customer and there is nothing which the customer can do about it. Although trials may help, the COTS licensing model essentially makes the customer pay up front before having gained any real experience of the technology or of the organisation which supplies it. Having been paid for an already sunk development investment, there is no direct incentive for the vendor to better match the product to the emerging customer requirements. Furthermore there is no guarantee that the supplier will not lose interest and honour the letter of the contract but fail to deliver the value which the customer wants.

OpenNMS report that their customers have typically been using the product long before they approach the team for commercial development or support. This means that they already have some understanding of the project and opinions about what they need. Similarly, since OpenNMS group only get paid for services, they are highly motivated to work with the customer to ensure that any paid-for development both improves the product and meets the customer's needs.

In trying to understand the OSS market, the researcher has found that the OSS industry is very closed for anyone in an unprivileged position to gain an understanding of the detailed functions performed by commonly used COTS OSS components. Everything is shrouded in a degree of marketing secrecy. One of the University Supervisors of the undergraduates

---

[52] BEA donated XMLbeans to the Apache Software Foundation in September 2003 but continue to use it internally http://dev2dev.bea.com/xml/xmlbeans.html [Accessed Sept 2007]

supporting the Catalyst visited TMW during the 2005 Catalyst and noted that it was impossible to tell from their marketing literature what functions any of the OSS components on display actually performed.

Whereas the generic middleware vendors are now going to great lengths to help developers understand their technology - increasingly through open source implementations (see Section 3.4.3.3), the OSS vendors have not followed the same path. One can generally only get to understand their products by being a customer (or prospective customer) or by paying a great deal for training.

In OpenNMS open source allows the customer to evaluate all of the claims of the project; both in the current code and the future direction which the project is executing. The customer is also able to directly evaluate the skills and ability of the key contributors. If additional functionality is required, the customer can make their own estimates as to the effort this will take and work with the open source vendor to price and realise a solution.

We noted above that open source encourages honesty in the relationship between the supplier and customer. This of course assumes that the power relationship favours the vendor. In reality, however most OSS vendors are small and their service provider customers exhibit huge market power. Code concealment is one way for suppliers to maintain their position. ISV's fear that open source could allow the customer to free-load and take advantage of the project without offering any return. OpenNMS has 1000 complete downloads a month - but the project has no idea where the majority of these are being used. In investigating for OpenOSS the researcher came across at least one example where a large telecoms company used OpenNMS but saw no need to contribute back to the team. In one sense this is not a problem - since the more people using the product, the more valuable it becomes. However (*Norris, J. S. et al. 2004*) of NASA recommends that it is good practice for all users to at the very least send an email to the project lists describing the use being made of the project.

However the issue becomes more noticeable in the product evaluation stage. As noted in Section 4.2.2.1, large service providers have come to expect vendors to support product evaluations and trials out of 'cost of sales' and expect to influence the roadmap of vendors who have made it to a 'preferred supplier' status. Even within the TMForum, there is the expectation from the service providers that they should not have to pay for work to be done. This market stance does not favour open source projects.

Finally, developers need confidence that the work which they contribute will be accepted by the project. (*Mannaert, H. et al. 2005*) experienced a number of difficulties in contributing their enhancements back to the open source projects they used to develop their Cast4all system (Section 5.8.1). However they surmise that the projects which they were enhancing (Cocoon, JOnAS and Tomcat) were general purpose components with large and diverse user bases and

possibly preferred to keep their independence from each other. They also felt that the proposed enhancements were too specific for the majority of users.

The researcher has found that OpenNMS have been receptive to his contributions. However this has followed from a concerted effort to meet with the team and explain to them the motivation for the work. The team were receptive because addressing Telecoms OSS was an aspiration for the project. The key observations are firstly, that it is very important to build good working relationships within an open source project which involve all of the stakeholders and secondly, that it is important to understand and ideally be aligned with the long term motivations driving the development team.

Chapter 7
Discussion and Conclusion

Chapter 7 Discussion and Conclusion

This final chapter will conclude by summarising the Thesis and reviewing the overall impact of the research program. It will consider whether one can draw together more general lessons from the research and will provide some recommendations for the players in the OSS industry including a discussion of possible directions for future work.

7.1 Summarising the Thesis

Chapter 1 set out the case for embarking on this investigation by collating evidence that the Telecommunications OSS industry has had an unfortunate history of trying to solve the endemic problem of poor OSS systems integration. Several initiatives and technologies such as CMIP, CORBA and TINA-C have failed to live up to their initial promise. The case was made that one reason why these standards failed to deliver was that they were not proven and disseminated through reference implementations (ideally open source).

A case was also made that some industry experts were concerned that academic research into network management was often uninformed by real industry experience and that the results were either unrepeatable or far removed from practical implementation. This led to the interesting question of whether Action Research could be used to allow academic researchers to provide more positive support to the industry as it struggles to implement OSS solutions.

The research program set out to discover whether open source might provide a vehicle to address the fundamental OSS integration problems, improve the standardisation process and, combined with Action Research, allow the results of academic research to be more relevantly disseminated into industry.

In Chapter 2 a case was made for attempting to use Action Research as a methodology to both learn about, work with and in some way change the practices of the telecommunications industry. It was hoped that Action Research would provide an academically rigorous framework which would allow the researcher to experience first hand both the business and technical issues of applying open source to the OSS problem. A proposed research methodology (Mr) was elaborated, drawing on the experiences of various researchers who are pioneering the use of Action Research in the field of Information Systems.

In Chapter 3 the OSS Industry and Open Source Problem Space (P) was analysed in order to draw out the key question which needed addressed through the planned Action Research interventions. An analysis framework was proposed based upon the notions of Risk Efficiency, The Component Market place and Collaborative Innovation in order to understand where and why open source might benefit the industry. It provided some insight into the TMForum' s role as a non-market institution which reduced the risk of New Product Introduction and suggested that open source could provide a complimentary institutional framework which increased the flow of 'sticky' information' between organisations.

Although this analysis raised more issues than could realistically be addressed by the rest of the program it articulated a very useful set of question which managers in the OSS industry should ask when considering whether to use open source. Although based around the OSS industry, it provides a useful template by which the same process of analysis could find applicability in other IT segments. The analysis in Chapter 3 informed the objectives of the three major interventions described in the following chapters.

The focus of the investigation in Chapter 4 (OpenOSS in the TMForum) was on the potential for using open source to further the collaborative innovation conducted between the key stakeholders in the OSS industry under the auspices of the TMForum. By working with major industry players on a short but very public open source exercise, the research exposed many of the issues which a more concerted open source OSS project would need to address. This work lead directly to the NGOSSoil intervention which exposed some of the systemic attitudes in the industry which presently prevent the wider use of open source to assist with standardisation and solving integration problems.

The focus of the investigation in Chapter 5 (DVB-T Service Management) was on understanding how open source could further the dissemination and application of emerging OSS standards in a novel problem space. Action Research provided a framework to analyse the problem with the project participants and the SANRR methodology (Table 32), recommended by the NGOSS lifecycle (*TMForum 2004a*), was applied to begin to characterise the problem using the SID (*TMForum 2004f*), the TAM (*TMForum 2006a*) and the recommendations of the SLA Management handbook (*TMForum 2004e*).

The TMForum have been encouraging members to use NGOSS as a toolkit to help them build their OSS systems. Although the NGOSS lifecycle provides some guidance, there is no one accepted way in which these tools can be applied and the TMForum has been actively calling for members to submit use cases. This case has applied NGOSS to the problem of managing DVB-T networks and has demonstrated how potential users of the NGOSS standard would also benefit from an open source implementation toolkit to experiment with ideas rather than having to rely on a paper specification with some UML models. The proof of concept has been instrumental in informing the specification of a system to manage the forthcoming UK DVB-T network being built by Arqiva.

In Chapter 6 (Contributing an OSS/J Interface to OpenNMS), Action Research was used to analyse the experiences of the OpenOSS Catalyst members as they used an established open source network management project, OpenNMS (*OpenNMS 2007c*) in a Glue-Code COTS Based Application (GCBA) project (Table 17). The researcher also used Reflective Learning as part of the Action Research process to extract lessons from the personal experience of becoming an accepted contributor in the OpenNMS community. Whereas much research into Open Source is done through external observation, this Action Research intervention took the

researcher along the same path which a telecoms adopter of open source would have to follow in order to become part of an open source project.

The intervention provided a case study of the practical problems faced by an ISV in establishing a commercially viable open source project and noted the fundamental differences between an open source business model which is based on sharing information and the COTS model which is based on concealing information. It proposed certain behaviours which the open source model requires of its participants to be successful (Courage, Conviction and Faith -Section 6.5.4).

Chapter 6 also referenced the OSSbeans project (*OSSbeans 2007*) which was created to prototype the practicalities of using open source to assist developers in ISV's more rapidly implement NGOSS compliant interfaces. OpenNMS was enhanced using OSSbeans to create a component which met the requirements for an innovation toolkit for the OSS industry (based on von Hippel's work) outlined in Section 3.6.3 and Table 25. The utility of this toolkit was proven in the DVB-T project previously described in Chapter 5.

Having reviewed the individual interventions in previous chapters, this concluding chapter will consider whether one can draw together more general lessons from the overall program by reflecting first upon the impact of open source on telecoms OSS (Mps) and secondly on the Action Research process (Mr). This broadly corresponds to the Action Research Specifying Learning stage (*Susman, G. I. et al. 1978*).

7.2 Reflections on the impact of open source on telecoms OSS (Mps)

7.2.1 NGOSS and Open Source

Section 3.5.1.3 suggested that NGOSS was a strategic attempt to reduce the risks associated with COTS based development and that open source could further help in managing the risks. Chapter 5 has shown how Arqiva were able to reduce the risks of developing a DVB-T management solution through a proof of concept using open source implementations of NGOSS like components. Chapter 6 has also illustrated a process by which a COTS ISV could be practically helped to migrate towards NGOSS by using open source NGOSS components.

Thus this work has proven that open source can provide significant risk reduction both by fostering collaborative innovation between complimentary products and by the dissemination and implementation of standards. It has also provided evidence that open source could provide a mechanism for service providers to re-engage with both OSS standards and with their vendors' New Product Introduction processes thus addressing some of the issues with the standardisation process outlined in Section 1.5. However this analysis may also have revealed a basic contradiction in the overall NGOSS strategy.

Section 6.5.4.3 observed how working with OpenNMS allowed the researcher to experience at first hand the contrast between the very restricted product information generally available across the COTS OSS industry and the detailed product information available for OpenNMS as an open source project.

The stakeholder analysis in Appendix D.4.3 provides evidence of the strategies generally being used by the service providers to overcome this lack of public product information. They can gain a degree of architectural understanding by creating 'walled gardens' within their organisations in which they bring together their chosen component vendors through privileged contractual relationships and under NDA's. This was evident in BT's OSS rationalisation strategy which is to work with an approved list of vendors (SALSA) to make their components integrate with BT's Systems Capability Framework - through product modifications or by using integration hubs as wrapping technology. BT's approach is consistent with the industry's NGOSS migration strategy discussed in Section 3.5.1.3 and it was noted that over time BT would like their Systems Capability Framework to be standardised as NGOSS Contracts which OSS vendors would be encouraged to implement.

A similar, though less permanent, approach is evident in the TMForum Catalyst program which creates multiple but much more limited 'walled gardens' in order to provide a vehicle for testing interfaces between OSS or equipment vendors. One leader of a 2007 Catalyst[53] told the researcher that he had included his developers in the Catalyst precisely because it gave them visibility of other OSS systems which they would not otherwise have been able to obtain. However the researcher could find no useful published results from previous Catalysts to guide the OpenOSS implementation. Participants within the Catalysts may learn technical details about each others products but little information, beyond marketing material, is published to help outside observers.

The fundamental problem observable here is that nobody outside of commercially privileged relationships can get access to useful information about OSS products - importantly this includes potential developers of complementary products. The cost of information is very high and this significantly slows down innovation across the industry.

Traditionally NGOSS has been characterised as an attempt to solve the problem of too much information by rationalising the concepts found in multiple OSS's onto a common architecture and nomenclature (like the TAM (*TMForum 2006a*)). However, one can also argue that NGOSS is an attempt to overcome the problem of too little information caused by the unwillingness of vendors to reveal details of their products.

---

[53] Interview recorded with Nigel Davis Joint leader of Nice 2007 Catalyst 'Harmony - Converging Standards to Integrate Emerging Technology', Telemanagement World May 20-24 2007, http://www.tmforum.org/Page32205.aspx [Accessed Sept 2007]

Appendix D.4.4 observes how service providers would like to use MDA techniques to design OSS systems using components which expose NGOSS Contracts. Researchers in the FORM project, (*Lewis, D., Wade, V., & Cullin, B. 2003*) have provide some useful insight into the ongoing thinking behind NGOSS Contracts as a means to exchange information between stakeholders in the design. They identify the same stakeholders in OSS development (Service Provider's, ISV's SI's and Standards Bodies) which were identified in Chapter 3. Their work aims to support the technology neutral modelling and assembly of OSS components into OSS systems by providing;

> ... a simple, minimum set of modelling guidelines that supports the exchange and comprehension of models between organisations involved in the development of integrated component-based OSS ...so that the differences in business contexts and behavioural semantics that present the greatest obstacles to integrating component models from separate sources can be readily identified and addressed early in the application assembly process. (*Lewis, D. et al. 2003*)

In short they are proposing to use NGOSS Contracts to address the same problems of communication between developers which this research has noted can also be significantly helped by open source (Section 6.5.3)

The NGOSSoil case highlighted the TMForum's present inability to strategically adopt open source as a tool to aid the furtherance of the NGOSS standard. This was partly explained by two factors. Firstly by the stakeholder observation in Appendix D.3 that the TMForum is an agent which seeks to maximise revenues from its membership - the majority of whom are COTS vendors and do not yet welcome open source. Secondly by the observation in Section 7.2.2.1 that service providers are not yet ready to engage with the open source process.

Martin Fowler, author of UML distilled (*Fowler, M. 2004*) has suggested that the notion of a Model Driven Architecture having a Platform Independent Model is a malapropism (*Fowler, M. 2003*). One is merely substituting one technology (UML, MOF, XMI etc) for another (Java, .net, C++ components etc). In the case of NGOSS, the TMForum appears to be substituting an NGOSS Contract for the revelation of the behaviour of a component at the code level. Having substituted one set of technologies for another, the industry may be no further forwards in using the standardisation process to solve the OSS integration issues discussed in Section 1.4. In fact, being technology neutral, there is presently no direct feed back loop one can use from implementation to prove that the NGOSS standard is correct.

If one was to take an extreme position one could therefore argue that that, in the case of NGOSS, the TMForum and the service providers are avoiding the real cause of the problems of integration between COTS products - the lack of information sharing and inflexibility inherent in their supplier ISVs' business models. Instead, by heavily promoting NGOSS as a technology neutral standard, the TMForum have been able to keep the ISV's content because none of their products actually expose NGOSS components. NGOSS is substituting a technology in which few members have a direct stake (and therefore few are directly threatened) for direct engagement with the underlying components. The only place where this

strategy has caused a problem with the membership is where it has been difficult for the tool vendors to agree on a standard by which one tool can handle both eTOM and UML models and thus potentially support NGOSS MDA (Appendix D.4.4).

The intent of this analysis not to rubbish the NGOSS standard or the many benefits which service providers have derived from it and its derivatives OSS/J and MTOSI. Indeed Chapter 5 showed how useful NGOSS and OSS/J can be in helping to solve a novel service management problem. Neither is the intent to dismiss the very important ongoing work being done on NGOSS Contracts which, when completed, will provide a very useful tool for using MDA to abstract the design of OSS systems. The intention is, however, to suggest that the OSS integration problem is primarily a management strategy problem and not a technical one. One is led to question whether NGOSS alone is a viable approach to solving such a difficult industry problem.

From a management strategy perspective, the current NGOSS approach appears to favour what (*Mintzberg, H., Ahlstrand, B. W., & Lampel, J. 2001 p.47 ff.*) have termed a 'planning approach' to strategy i.e. a top down strategy whereby the 'architectural gurus' create a specification which they try and impose on an existing industry or organisation. It is limited by the bounded knowledge of its creators and the willingness of the industry to adopt the standard.

Open source facilitates what (*Mintzberg, H. et al. 2001 p. 175 ff.*) would term a 'learning approach' which promotes strategy formation as an emergent process. In this model, the world is too complex to allow strategies to be developed all at once as clear plans or visions. Hence strategies must emerge in small steps, as an organisation adapts, or 'learns'. In the open source context, anybody who wants to contribute to the design of a system can do so by understanding the context of their own component and adapting it to the components which surround it. Encouraging open source implementations which feed back into NGOSS would be a more effective way to ensure that as many use cases as possible contributed to the standard.

In an economic study of standards, (*Ghosh, R. A. 2005a*) cites network effects, similar to those attributed to the OSS industry in Section 3.4.1. He suggests that one approach to network effects is to try to abstract the network externalities from specific products by identifying the features of the technology that provides the network effect and ensuring that its use is not limited to one product or service. Thus products from different ISV's are made interoperable by agreeing to open standards for the basic technology components that provide the network externalities. In theory this allows a natural monopoly to arise in terms of the technology, but competition to thrive in terms of the actual products and services that interoperate. This approach gives an economic context to the industry migration path towards NGOSS discussed in Section 3.5.1.3.

(*Ghosh, R. A. 2005a*) argues that to be successful this strategy requires reference implementations to augment - if not, perhaps, replace - the formal specification of the standard. He goes on to suggest that when such a reference implementation is available under an open source licence, it may achieve the same economic effect as an open standard, even without the institutional processes of standard setting. This is because the reference implementation may act as the formal specification (especially if sufficiently well documented) and be reproduced without economic restrictions by any potential vendor of the technology. In other words (*Ghosh, R. A. 2005a*) is making a case which suggests that the OSS industry should think very seriously about creating some basic open source NGOSS components (enabled through its OSS/J and MTOSI derivatives) in order to boot-strap the network effects which would lead to the wide spread adoption of NGOSS component systems.

This is the basic philosophy which has underpinned the creation of the OSSbeans project and contributing an OSS/J interface to OpenNMS. Although still very incomplete, it shows what can usefully be achieved if the industry accepts the need to create open source NGOSS components.

If this basic argument is accepted, then the OSS industry stakeholders would need to change their practises to support open source. The following sections suggest ways in which practices would need to change.

7.2.2 Impact on practice

In line with the framework laid out in Table 10 and Section 3.3.1, this section will complete the analysis by recommending some 'Impacts on Practice' for OSS industry players who are considering how to respond to the open source phenomenon.

7.2.2.1 Service Providers

This research has shown how the strategic use of open source is already becoming a significant factor in competitive advantage on the internet. Section 4.2.4.2 discussed how the open source scenario envisaged in the OpenOSS Catalyst was being implemented in the low cost business models of companies like Truphone. Section 3.4.2 discussed how open source enabled a culture of rapid innovation in companies like Google operating above the IP demarcation. This ability to support rapid innovation using open source contrasted with the incumbent service providers who, having outsourced their R&D, were relying on a troubled standards creation process to guide long term COTS development (Section 1.5).

Section 3.4.3.3 described how major middleware vendors were beginning to embrace open source as part of their business model which (*Silva, N. et al. 2006*) has shown could be used to build realistic NGOSS solutions. Appendix E.2 illustrates how, even as an embryonic open source OSS system, OpenNMS is already attracting users from across the Telecommunications Industry layer model. The OSS/J implementation work and DVB-T trial (Chapter 5 and

Chapter 6) have demonstrated that open source OSS implementations could viably be taken further.

The above analysis suggests that the traditional service providers are being exposed to aggressively innovative competition enabled in part by the strategic use of open source. Strategically service providers should be looking at participating in open source projects both for cost reduction and in order to regain more direct control over the implementation of innovative technology. Yet the NGOSSoil intervention (Section 4.3.4.3) exposed the unwillingness, without an immediate business case, of any members of the Service Provider Leadership Council in the TMForum to support, what appeared technically to be a useful open source NGOSS component.

Perhaps this can be explained by the fact that the present generation of OSS managers have staked their careers and developed personal expertise in the incumbent set of OSS products. Having created 'walled gardens' of preferred suppliers, there is no incentive to look for outside solutions. However Antonio Rosso of the OpenNMS project has observed that (at least in Italy) a new generation of managers are beginning to emerge in telecoms who are more favourably disposed towards open source[54]. As they come into positions of influence, how should they change the stance of the service providers in order to embrace open source?

We suggested in Section 3.3.2 that the business case for major infrastructure investments such as moving to NGOSS might be better made using a Real Options approach to valuing the platform rather than trying to justify it using an ad-hoc selection of projects. Although the wide spread adoption of Real Options to financially justify projects might be too big a step for a public company to make, it is still within the bounds of a CIO or R&D organisation to apply this approach to planning discretional expenditure on open source projects. This is especially so if open source participation is believed to provide a more effective means to influence COTS product direction than standards development alone and the aim of R&D is to 'learn how to use technology' - as described in Section 3.4.2.

Having established a basis for justifying long term engagement with open source projects, the service providers should set about building a competency and budget for open source project participation. This should draw together developers in R&D and end-users such that there is a coordinated approach to making sure that open source participation is as close to end user use-cases as possible. This should facilitate the user based innovation, market adaptation and NPI processes discussed in Section 3.6

Since the service providers have previously rewarded skills in using COTS OSS products they need to address the inertia this causes against moving from incumbent COTS systems by rewarding open source talent. Quick win projects should be sought to build the internal

[54] Private Interview with Antonio Rosso at OpenNMS Dev-Jam August 2007 (on tape)

awarcness of the value of open source and to indicate to the organisation that open source participation will be rewarded as much as competency in COTS OSS systems.

Much of the industry expertise now lies with the OSS vendors. Therefore any service provider strategy for using open source must embrace the existing OSS suppliers and leverage their existing long term contractual relationships. Service providers should begin by making it clear to their suppliers that they would look favourably on opportunities to become more involved in the forward development of their OSS products by using open source. Service providers will need to demonstrate however that their objective for using open source is not to put the current OSS suppliers out of business but rather, as discussed in Section 6.5.3, to increase the value of their products, increase innovation, force the pace of NGOSS standards adoption and reduce the integration tax. The service providers must demonstrate that they are acting in good faith as discussed in Section 6.5.4.3.

Since the middleware vendors are already moving quickly in the open source direction, the service providers should begin building their internal open source competency by engaging directly with relevant middleware open source projects - as a potential end user. Small but strategic projects should be sought which build competency and deliver value to the organisation. These should be used to signal to the rest of the OSS industry that the service providers are serious - and at the same time to encourage the ISV's to begin looking seriously at using open source middleware stacks in future developments.

Finally, the service providers should begin to look at addressing their organisational barriers to the use of open source including security issues raised in Section 6.5.2.2. Some suggestions about which processes will need looked at by service providers based upon the eTOM are included in Appendix C.

7.2.2.2 ISV's

We noted in the stakeholder analysis in Appendix D.1 that one of the reasons for Agilent's participation in the OpenOSS catalyst was to assess the danger and opportunities represented by open source as a disruptive OSS technology (*Christensen, C. M. 2003*), (*Christensen, C. M. 2004*). Christensen's basic argument is that a disruptive technology may enter the market and provide a product which has lower performance than the incumbent but which meets or exceeds requirements at a price attractive to certain segments, thereby gaining a market foothold. The incumbent responds by moving up-market to address more attractive customers but eventually the disruptive technology will improve - ultimately becoming good enough to drive out the incumbent. This was observed beginning to happen with OpenNMS's high end customers in the case study in Appendix E.2.

However Section 3.4.1 reported evidence that an ISV may be able to co-exist with an open source substitute if it changes its behaviour by appropriate pricing, reducing the difficulty customers have in developing custom features, becoming more responsive to bugs and

reducing the integration tax. Again, judicious use of open source could help here - either through projects after the fashion of OSSbeans which tries to separate out the integration problem from the core product, or through duel licence models facilitating an open source development stream in which customers can participate.

ISV's should also address the issue of 'demand side learning' by which sophisticated user/programmers can easily learn how to use the software and modify it for their own purposes. The analysis of OpenNMS was able to observe practically how the 'demand side learning' works through customer innovation (Appendix E.5) and through the open knowledge and relationships fostered through the project (Section 6.5.4.3). This suggests several steps which an ISV could take towards addressing the challenges posed by open source.

Firstly, ISV's should look carefully at open source as a means for cost reduction. Is the current product expending efforts sustaining functionality which could be outsourced using an open source framework like Spring? Does it use closed source components which could optionally be replaced with open source substitutes? For instance, could it design in the options to substitute its operating system, J2EE application server or database with an open source equivalent to allow more pricing flexibility towards its customers?

With a closed source product, there is often not the incentive to re-factor unless there is a pressing business need. However, an ISV could significantly benefit from reducing the cost base of sustaining the product if it moves to a widely adopted framework with good documentation and an accessible supply of trained developers. OpenNMS is essentially minimising its cost base by taking this approach (Section 6.5.2.3). Even if the sub-components used are not yet licensed as open source, the ISV should ensure that the sub components they adopt have a strategy for responding to the challenge of open source.

Secondly, ISV's should look for features of their product which could exist as separate open source projects and thus benefit from wider adoption and potentially a larger development community. This mirrors the approach of BEA, Google and OpenNMS as discussed in Section 6.5.4.2. It also maps onto Nortel's motivations for proposing NGOSSoil discussed in Section 4.3.4.2.

Thirdly, ISV's should where possible combine their use of open source frameworks with their standards activities in new product development. This is a significant opportunity for long term cost reduction and benefit from interaction with other ISV's. As part of the process of moving towards using an external framework, ISV's should also seek engagement with service provider customers who are willing to participate in the framework's development (as discussed above in 7.2.2.1). However, as was observed in the case of NGOSSoil, this strategy will only really deliver benefit as a bridge to customers if they are also willing to engage in project participation for this purpose.

Finally, ISV's should explore different models for sharing value and risk with the service providers, perhaps using some of the techniques discussed in Section 3.3. The COTS licence model is becoming less attractive as networks become larger and the up front licence fee fails to reflect the time-value of revenues generated by new projects. Elaborating suitable mechanisms is beyond the scope of this present work but a starting point could be the suggestions of (*Cooke, C. 2006*) for using the newly created UK Limited Liability Partnership to provide a mechanism for revenue sharing based upon the beneficial use of code vested in an LLP dedicated to each project.

7.2.2.3 Integrators

The future role of integrators was of particular interest to SI's like Invocom, the sponsor of this project. Section 3.5.1.2 suggested that Systems Integrators typically use the Glue-Code COTS Based Application model to create end user systems out of components and noted the SEI definition of COTS based system development as '*occurring through the simultaneous definition and trade-off of the COTS marketplace, system architecture, and system requirements*' (*Brownsword, L. et al. 2000*). Section 3.5.1.3 suggested that NGOSS components were likely to be smaller and require more assembly with complementary components than current COTS packages. SI's will thus have an increasing role in identifying the types of components which are available and in finding ways to apply these to solving customer problems.

Section 6.5.2.3 described OpenNMS is a productised assembly of components and observed that the introduction of component frameworks was freeing their developers to concentrate increasingly on application code. Appendix E.1 also showed that the OpenNMS business model, like many open source ISV's is based upon a hybrid of support services and incremental development.

One can thus suggest that the application of the open source model blurs the distinction between Component Developers, Application Assemblers and Customers (*Vitharana, P. 2003*) within the OSS Component Market Place (Section 3.4). All three groups now need to be much more aware of the very rapidly changing component market place and consider the question; 'what new components or frameworks are available to exploit in my business?'

As was shown in Section 6.5.1, the focus of open source ISV's like OpenNMS will be on ensuring that the components they develop are well designed, are leveraging the best underlying frameworks and that they incorporate user contributions in a well structured manner. SI's will increasingly be their partners in getting to market.

The basic focus of the integrator will be on finding potential solutions to customer problems. While the preference may be to use COTS solutions, they will still need the capability to write code in order to exploit increasingly capable middleware technologies such as portal servers in one off solutions. However they should constantly be looking at whether solutions could be

spun off as a separate open source projects. In addition, they should be fostering long term relationships with strategic open source projects such that they can quickly adapt and apply them to new customers situations.

One of the issues observed by Tarus Balog[55] was that SI's currently make an easy living by reselling and marking up licences from COTS products. This makes them less interested in open source. As with the previous discussion on ISV's, SI's should begin thinking about brokering revenue sharing arrangements with customers which allow value to flow through to the underlying open source projects in order to provide a sustainable product solution.

7.2.2.4 Equipment Vendors

We saw in Section 3.4.3.1 that the Telecoms Equipment Vendors are already adopting open source in product development. In addition, equipment vendors like Nokia are following Apple iTunes in providing content services as well as terminal equipment (*Le Claire, J. 2007*). As with the service providers discussed above, this pushes them above the IP layer and exposes them to competition from parties already familiar with open source.

The discussion of NGOSSoil in Section 4.3.4.2 observed that equipment vendors do have reasons to promote open source OSS integration solutions in order improve the uptake of their equipment. In Section 1.3.5 we saw an increasing need for the equipment vendors to integrate their models more closely with the OSS vendors because of increasingly intelligent network equipment.

This very much points to the value of open source as an integration tool but it also points to the need to rethink the revenue model associated with supplying OSS systems to support network equipment. At present there is a hard demarcation between these markets however with more intelligent networks, it is unlikely that the OSS vendors will still be able to command such a high licence fee for each device managed. Again open source could partly solve the problem by allowing the OSS vendors to reposition their development in order to take part of their revenue from the end customer and part from the equipment vendors by providing OSS middleware which offers value to both stakeholders.

7.2.2.5 Standards Bodies

Through participation in the TMForum's activities, the researcher has been able to observe first hand the value which the TMForum adds to the industry. An important contribution of this study has been to articulate and demonstrate in practice the role of the TMForum as both a standards body and non market institution (Section 3.6.2). Section 1.4 observed that the TMForum was originally created in order to bridge the gap between 'official' standards organisations and the market place. Over time it has taken on more of a role in the creation of standards and now contributes to the more formal standards bodies like the ITU-T. Thus the

---

[55] Private Interview with Tarus Balog at OpenNMS Dev-Jam August 2007 (on tape)

TMForum may represent an emerging model of an organisation which exists to foster practical market based collaborative innovation rather than just create standards.

Section 1.5 suggested that the thoughtful application of open source may represent a key strategy for standards development organisations like the TMForum to maintain their market position. Although Section 7.2.1 presented a fairly harsh assessment of the TMForum's NGOSS program, this was with a view to emphasising the problems which the industry will have in introducing NGOSS without an open source strategy. Promoting such a strategy will require the TMForum to take several practical steps.

Firstly it should establish an open source developers' forum where developers within the service providers can compare notes on their present use of open source technologies to create OSS solutions. In Section 3.4.2 we saw that some 200,000 IT professionals are directly employed by the service providers and so the TMForum should consider providing a mechanism for engaging directly with some of them about open source. Developers from the ISV's who are already promoting open source components should also be encouraged to attend. The TMForum's open source program which was a direct result of this Action Research activity is a start (*TMForum 2007d*).

Secondly, rather than try and develop an in house competency in open source project management, the TMForum should foster relationships with organisations who are already experienced in creating open source projects in a telecoms context. These could include the France Telecom sponsored ObjectWeb Consortium (*OW2 2007b*) and The Open Group (*TheOpenGroup 2007c*) who developed the OpenPegasus implementation of the DMTF's WBEM standard.

Thirdly, the TMForum should try and encourage the Universities who are presently researching NGOSS issues to begin looking at creating open source components rather than keeping their work in house. This could happen under the auspices of the TMForum university program (*TMForum 2007e*).

Finally, and perhaps most importantly, the TMForum Board and particularly the Service Provider Leadership Council needs to reach the conviction that embracing open source is a necessary next step for the evolution of the NGOSS program and the needs of the OSS industry. As was amply proven in the NGOSSoil experience (Section 4.3) and as was discussed in Section 6.5.4.2, without senior sponsors who have both the conviction and resources it will not be possible to make an open source program a reality.

7.2.3 Risk Efficiency and Open Source Introduction

Risk efficiency as articulated in Section 3.3 has been used as a qualitative notion to assist in analysing the market and understand the projects within this Thesis although the full application of risk efficiency to open source project management in accordance with the

techniques described by (*Chapman, C. B. et al. 2003*) has not been attempted. The projects were too short to benefit from this exercise. However it would be a very useful next step to use Action Research to test the utility of these concepts in a large scale open source project in the future.

In a similar vein, the notions of using Real Options to justify investment in open source and/or NGOSS platforms as suggested in Section 3.3.2 have been used to heuristically address some of the issues encountered in this study but a full financial analysis of an open source project using Real Options has not been attempted. This would also be a useful further line of study.

(*Kwan, S. K. & West, J. 2005*) have proposed a Conceptual Model for Enterprise Adoption of Open Source Software which is intended to test the IT buying behaviour of large firms. The model (Figure 25) mirrors in many respects the analysis in this Thesis and is based upon a three phase process to be used when choosing COTS software; 1) Determine a software adoption policy; 2) Search for a candidate set of suitable products; 3) Choose a particular product. The experiences from implementation should then be fed back to influence future policy.

Their model and framework makes explicit trade-offs between features, risk, and cost and notes how these trade-offs vary for different systems across the IT portfolio. They note that each system has a different strategic importance as characterised in Table 39 and that a system's importance is related to the firm's strategic response to the industry context. They also note that a system's strategic importance can vary with time. This mirrors discussion in Section in 3.3.1 where it was noted that a niche telecoms project in the early adopter phase may be able to bear more a risky implementation strategy than a mainstream service.

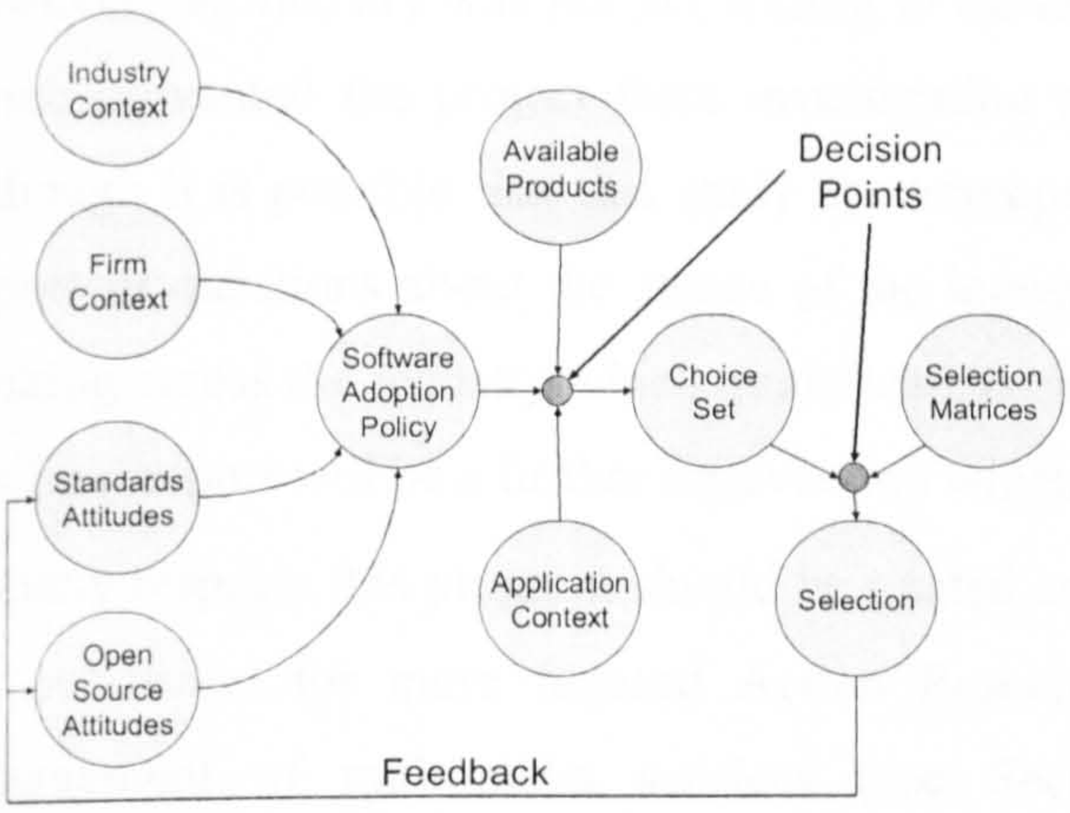
(*Kwan, S. K. et al. 2005*) set the expectation, drawing on the work of (*Dedrick, J. & West, J. 2004*) and their own interviews, that attitudes towards open source in a firm should roughly parallel those towards open standards. However this Thesis has shown that although there has been a long standing commitment to creating (if not always adopting) standards, the OSS industry has had a low adoption rate towards open source when compared with other sectors. However, should open source OSS systems become available, (*Kwan, S. K. et al. 2005*) may offer some hope that they would be taken up by the industry.

The chief difference between this Thesis and (*Kwan, S. K. et al. 2005*) lies in the fact that they are comparing individual complete COTS and open source products rather than multiple integrated systems. They have not considered the issues of development, integration or product roadmap management discussed in Sections 3.5 and 3.6.

(*Kwan, S. K. et al. 2005*) recommend describing the industry context using Porter's 5 forces (bargaining power of suppliers, substitutes, potential entrants, buyers (*Porter, M. E. 1980*)). Although Porter's analysis is a well known management tool, it essentially provides no more information than the analysis done of the OSS Component Market Place in Section 3.4. What

it doesn't provide is a direct mechanism to link the analysis with a project management process as described by (*Chapman, C. B. et al. 2003*). The analysis presented in this Thesis has taken this much further by introducing the concepts of risk efficiency as proposed by (*Ward, S. et al. 2003*) in order to allow a project manager to better quantify and manage the uncertainty surrounding the choice of a portfolio of COTS and open source products.

In a similar vein, the Eurescom work of (*Pereira, J. P., de Matos, V., Chainho, P., Lin, G., & Mäkinen, S. 2001*) and (*de Bruijn, B., Schickentanz, R., Grimm, L., Wilhelm, U., Chainho, P., Vogel, V. et al. 2001*) has provided some useful benchmarks for assessing open source components in a telecoms context as has the work of (*Bruce, G. L. et al. 2005b*) and (*Bruce, G. L. et al. 2005a*) which draws upon their involvement in the OpenOSS Catalyst. However these reports have concentrated on the cost reduction potential of open source rather than considering the potential value of open source for risk reduction, on-going standards innovation and product management which are discussed in this Thesis.



3 Phase process:

- Policies. The firm establishes a series of policies (formal or informal) that guide its procurement decisions.
- Choice set. From the range of available options, at the first decision point the firm identifies those acceptable alternatives that meet minimum requirements.
- Selection. From the range of acceptable alternatives, at the second decision point, the firm makes its final selection decision.

Figure 25 Conceptual Model for Enterprise Adoption of Open Source SW
(after Kwan &West 2005)

Table 39 Categorisation of Enterprise systems (Kwan &West 2005)	
Laboratory.	These non-production systems are developed in response to demands for pilot studies and experimentation with new technology.
Support.	These 'routine systems' provide business value by improving the enterprise's internal efficiency;
Mission Critical	The smooth and reliable operations of the systems in this stage are critical to the fulfilment of the mission of the enterprise, and their failure (however temporary) can subject the enterprise to loss of sales, profits, and customer loyalty.
Strategic.	These are systems that provide actual (not imagined) competitive advantage over rivals.

7.3 Reflections on the Action Research Process (Mr)

The key contribution from this program has been as the first known comprehensive attempt to research telecoms OSS systems using the Action Research paradigm. Chapter 2 has already laid out the chosen Action Research methodology, the chronology of the interventions and key steps taken to maintain the quality of the work. This section will reflect on whether the Action Research process has added value to the program.

The problems besetting OSS development set out in Chapter 1 were very broad and yet of great importance to the Telecommunications industry. The many previous attempts by the industry to address these problems were mainly technology based but the reasons for failure did not just lie in the technology. This naturally raised questions which had exercised the researcher professionally for many years. Action Research potentially provided a research paradigm which could cross the business and technical disciplines in order to get to the underlying issues.

The participants entered the study with high hopes of influencing the industry towards using open source but as with many such initiatives, the impact has been less than hoped for. The first lesson for the action researcher must be that the influence of Action Research is ultimately limited by the willingness of the area of study to be influenced. The project benefited greatly in its early stages from the fact that open source was becoming visible to the industry and they wanted to know more. This would correspond to Rodgers early adopter stage (Figure 6). However the industry was not yet willing to move to a point of more general implementation which prevented the project from investigating a more complex open source OSS project. Although it is possible that this study has attempted to cover too much ground, it has raised important questions about the stance of the industry and more importantly has set in motion thinking across the industry which might lead to change. It is to be hoped that dissemination of this final report will be a further intervention which will provoke further action.

In many respects this program should be treated as an exploratory study which has opened up the possibility for more focused Action Research. Attempting Action Research into the management of multimedia services (see Section 4.4.1.2) and Action Research into information systems modelling using NGOSS (see Section 5.8.2) look like particularly promising areas. However Action Research has been applied to other areas which could also have rich application to the OSS space including the design of human computer interfaces (*Gulliksen, J. & Göransson, B. 2000*) and the design of security systems (*Page, V., Dixon, M., & Choudhury, I. 2007*).

In addition the TMForum has proven to be a good environment for university researchers wishing to engage in Action Research. This may also be true of similar industry forums (certainly the DMTF). Although a practical observation would be that not every student will have the entrepreneurial capability or industry knowledge to broker the necessary relationships. However for those researchers already involved in the TMForum university partnership, the application of the Action Research method described in Chapter 2 should make interesting reading.

Section 3.6.2 raised the conjecture;

> Conjecture: That the OSS industry might have a great deal to gain if it could effectively integrate the contributions of an active standards defining trade association with open source projects and university research programs.

Participation in OpenOSS allowed the researcher to become an active member of the NGOSS community and develop an open source project within the TMForum. This in itself was a significant contribution and will hopefully set a pattern for future projects.

The first observation would be that Action Research, as applied in this program, allowed the university searcher to engage seamlessly with the normal work of the TMForum members. As far as the other participants in the OpenOSS catalyst were concerned, this approach made the researcher appear identical to another commercial participant. The other participants were aware of the Action Research agenda but were not necessarily aware that the researcher was doing anything more than they were in contributing to the project.

The second observation would be that Action Research afforded the researcher the flexibility to define the research direction as the project proceeded. The process of defining and setting up the project was recognised as being as much a part of the research as performing the work tasks. This was important since this represented at least 50% of the work load. The fact that the research plans have had to change to match the opportunities available for the researcher to get access to projects is amply illustrated by comparing the very different forward plans envisaged in the 2 year transfer Thesis (*Gallen, C. R. & Reeve, J. S. 2005a*) with those elaborated in this final Thesis. As discussed in Section 2.2.4, action researchers need to be constantly developing a portfolio of options for taking their work forwards in case they get stuck.

The most satisfying aspect of this research program has been that it has not just been research for research's sake. The interventions arranged by the researcher have provided immediate tangible benefits for all of the participants. The researcher has personally learnt a great deal about the practical use of OSS standards and open source programming. Much of this learning has been tacit. The researcher would now feel quite confident in engaging with NGOSS and with an open source project. However as (*Esperanza, M. 2005*) notes there is a significant difference between pure software development and software engineering research. This study has tried to use the discipline of Reflective Learning within an Acton Research process to draw out more general lessons from involvement in the various projects.

The Action Research methodology of setting out a research epistemology in the form of a set of questions about open source in the OSS industry (Chapter 3) has been very helpful in shaping the project interventions. The industry analysis leading to Chapter 3 has been a process of setting out a portfolio of questions in order to see which ones could subsequently be investigated.

The astute will note that the researcher has not necessarily addressed all of the questions raised in Chapter 3. Instead Chapter 3 has provided a background against which to judge each intervention on its own terms. This may not appear entirely satisfactory but the plain fact is that the issues which could be addressed by each subsequent intervention were determined by

more than the researcher's agenda. There is plenty of scope for future interventions to fill in any gaps.

For the sake of tidiness one could have sought to limit the questions raised in Chapter 3 to only those which were directly addressed later - but this would actually remove value from the Thesis. Chapter 3 has raised questions as to the role of standards bodies like the TMForum. It has also raised important questions as to the ultimate trajectory of the open source revolution which is certainly spreading to telecoms from the IT industry. As such Chapter 3 alone will be of use to managers wanting to know what sort of questions to ask as they develop their own open source strategy within the OSS industry.

7.4 Future Work

Throughout this Thesis a number of areas have been outlined for potential future work.

Immediately, there is a need to complete the DVB-T modelling work in support of the UK's digital TV rollout and there is ongoing work which can be done with the OSSbeans framework to make it more attractive.

Having demonstrated the utility of Action Research in looking at the OSS problem space other researchers may wish to take up the challenge. There are numerous other areas in which it could be applied. These range from examining particular technology areas of interest to OSS such as Process Portals, User Interfaces or the utility of NGOSS modelling methods (discussed in Section 7.3) to further investigating the use of Real Options and Risk Efficiency in planning and executing commercial open source projects (Section 3.3).

However the researcher believes that the most important next step for this work is to begin operationalising the ideas contained in this Thesis through continued involvement in the TMForum's open source program and in the OpenNMS project. Disseminating the findings of this research through the TMForum will represent the final intervention of the project and may result in some further action as the SPLC consider further the issues it raises.

OpenNMS's future product plans potentially make it a vehicle through which open source NGOSS compliant components could be delivered to the industry - with or without the initial sponsorship of the service providers. OpenNMS is a commercially viable project and is now a member of the TMForum and so is in a position to pursue this strategy.

7.5 Conclusion

This thesis has summarised three and a half years of Engineering Doctorate research in the OSS industry. As an Action Research activity, the principal contribution has been to develop within the industry a greater understanding and acceptance of the role which open source can have in the delivery of OSS solutions. It has already had a significant affect on the TMForum's strategic direction, having lead to the creation of the TMForum's Open Source Program (*TMForum 2007d*) and contributed to the shaping of the Prosspero Program (*TMForum 2007g*).

This project would not have been possible without the support of many individual open source champions within a large number of organisations. Although open source has yet to make its full impact in the OSS industry, the existence of these influential individuals represents an encouraging groundswell of support which suggests that one way or another, open source will become an established tool for service providers. It is hoped that this work has made a useful contribution to furthering this goal.

References

References

1. (Adobe 2007)
   Adobe, "Adobe SVG Viewer", 2007, http://www.adobe.com/svg/viewer/install/ [Accessed Sept 2007]
2. (Agilent 2002)
   Agilent, "Agilent NgN Analysis System - Monitor and Manage next generation Voice-over-Packet (VoP) and Internet Offload Networks ", Publication number 5988-5974EN (Product Description), 2002, http://cp.literature.agilent.com/litweb/pdf/5988-5974EN.pdf
3. (Ainslie, B. J., Fisher, M. A., and Rana, S. P. 1999)
   Ainslie, B. J., Fisher, M. A., and Rana, S. P., "TINA - A Distributed Approach to Service Management", BT Technology Journal, 17, [2], 95-102, 1999, Kluwer Academic Publishers, Hingham, MA, USA ISSN 1358-3948, http://dx.doi.org/10.1023/A:1009642017741
4. (Alavi, Maryam and Carlson, Patricia 1992)
   Alavi, Maryam and Carlson, Patricia, "A review of MIS research and disciplinary development", J.Manage.Inf.Syst., 8, [4], 45-62, 1992, M. E. Sharpe, Inc, Armonk, NY, USA 0742-1222, http://portal.acm.org/citation.cfm?id=175403#
5. (Amidon Rogers, D. M.Amidon Rogers, D. M., "The Challenge of Fifth Generation R&D", Research Technology Management, 39, [4], 33-41, http://www.entovation.com/gkp/challenge.htm [Accessed Sept 2007]
6. (Anderson, R. 2005)
   Anderson, R. 2005. Open and Closed Systems are Equivalent (That Is, in an Ideal World). In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*: 128-142. Cambridge, Mass: MIT Press,.ISBN 0-262-06246-1
7. (AndroMDA 2007)
   AndroMDA, "AndroMDA Model Driven Architecture Framework Project Homepage", 2007, http://www.andromda.org/
8. (Apache 2007d)
   Apache, "Apache Log4J Project Homepage", 2007d, http://logging.apache.org/log4j/ [Accessed Sept 2007]
9. (Apache 2007a)
   Apache, "Apache Tomcat Project Homepage", 2007a, http://tomcat.apache.org/ [Accessed Sept 2007]
10. (Apache 2007c)
    Apache, "Apache Ant Project Homepage", 2007c, http://ant.apache.org/ [Accessed Sept 2007]
11. (Apache 2007e)
    Apache, "Apache Maven 2 Project Homepage", 2007e, http://maven.apache.org/
12. (Apache 2007b)
    Apache, "Apache OpenJPA Project Homepage", 2007b, http://openjpa.apache.org/ [Accessed Sept 2007]
13. (Apache 2007f)
    Apache, "Apache Software Foundation Homepage", 2007f, http://www.apache.org/ [Accessed Sept 2007]
14. (Arqiva 2007)
    Arqiva, "Arqiva", 2007, http://www.arqiva.com [Accessed Sept 2007]
15. (Avison, D., Baskerville, R., & Myers, M. 2007)
    Avison, D., Baskerville, R., & Myers, M. 2007. The Structure of Power in Action Research Projects. In N. F. Kock (Ed.), *Information systems action research: an applied view of emerging concepts and methods*: 19-41. New York, N.Y: Springer,.ISBN 9780387360591 (hbk.), http://dx.doi.org/10.1007/978-0-387-36060-7_2
16. (Avison, D, Gregor, Shirley, and Wilson, David 7-1-2006)
    Avison, D, Gregor, Shirley, and Wilson, David, "Managerial IT Unconsciousness", Communications of the ACM, 49, [7], 88-93, 7-1-2006, 0001-0782, http://doi.acm.org/10.1145/1139922.1139923
17. (Avison, D, Lau, F., Myers, M., and Neilsen, P. A. 1999)
    Avison, D, Lau, F., Myers, M., and Neilsen, P. A., "Action research", Commun.ACM, 42, 94-97, 1999, ACM Press, New York, NY, USA , http://doi.acm.org/10.1145/291469.291479
18. (Azmoodeh, M., Georgalas, N, and Fischer, SAzmoodeh, M., Georgalas, N, and Fischer, S, "Model-driven systems development and integration environment", BT Technology Journal, 23, [3], 96-110, Springer, Netherlands ISSN 1573-1995, http://dx.doi.org/10.1007/s10550-005-0034-1
19. (Baldwin, Carliss Y. and Clark, Kim B. 1-7-2006)
    Baldwin, Carliss Y. and Clark, Kim B., "The Architecture of Participation: Does Code Architecture Mitigate Free Riding in the Open Source Development Model?", Management Science, 52, [7], 1116-1127, 1-7-2006, http://dx.doi.org/10.1287/mnsc.1060.0546
20. (Balog, Tarus 2007b)
    Balog, Tarus, "Introduction to OpenNMS", Southern California Linux Expo SCALE 5X, February 10-11, 2007, 2007b, http://socallinuxexpo.com/scale5x/conference+info/speakers/Tarus+Balog/ [Accessed Sept 2007]
21. (Balog, Tarus 2007a)

Balog, Tarus, "Does OpenNMS have a Map?", OpenNMS FAQ page, 2007a, OpenNMS, NC , http://www.opennms.org/index.php/FAQ-About#Q:_Does_OpenNMS_Have_a_Map.3F

22. (Barth, W. 2006)
Barth, W. 2006. *Nagios - system and network monitoring*. Munich: Open Source Press, ISBN.1593270704

23. (Baskerville, R. 1999)
Baskerville, R., "Investigating information systems with action research", Commun.AIS, 2, 4, 1999, Association for Information Systems, Atlanta, GA, USA , http://www.cis.gsu.edu/~rbaskerv/CAIS_2_19/CAIS_2_19.html [Accessed Sept 2007]

24. (BEA 2007)
BEA, "SolarMetric Kodo JDO", 2007, http://www.solarmetric.com/solarmetric_jump_page/ [Accessed Sept 2007]

25. (Beck, K. 2003)
Beck, K. 2003. *Test-driven development by example*. Boston: Addison-Wesley, ISBN.0321146530 (alk. paper)

26. (Beck, K. & Andres, C. 2005)
Beck, K. & Andres, C. 2005. *Extreme programming explained - embrace change*. Boston, MA: Addison-Wesley, ISBN.0321278658 (alk. paper)

27. (Benkler, Yochai 2002)
Benkler, Yochai, "Coase's Penguin, or, Linux and The Nature of the Firm", The Yale Law Journal, 112, [3], 369-446, 2002, http://www.law.duke.edu/pd/papers/Coase's_Penguin.pdf

28. (Benkler, Y. 2006)
Benkler, Y. 2006. *The wealth of networks - how social production transforms markets and freedom*. New Haven Conn.: Yale University Press, ISBN.0300110561 (alk. paper) , http://www.benkler.org/wealth_of_networks/index.php/Main_Page

29. (Benoit, Hervé 1997)
Benoit, Hervé, "Digital Telivision, MPEG-1, MPEG2 and principles of the DVB system", 1997, Arnold, London ISBN 0340-69190-5

30. Bergquist, M. & Ljungberg, J. 2001. The power of gifts: organizing social relationships in open source communities. *Information Systems Journal*, 11(4): 305-320.

31. (Black, Richard, Donnelly, Austin, and Fournet, Cedric 2004)
Black, Richard, Donnelly, Austin, and Fournet, Cedric, "Ethernet topology discovery without network assistance", Proceedings of the 12th IEEE International Conference on Network Protocols, 2004.ICNP 2004, 328-339, 2004, IEEE, http:dx.doi.org/10.1109/ICNP.2004.1348122

32. (Blum, Fred H. 1955)
Blum, Fred H., "Action Research--A Scientific Approach?", Philosophy of Science, 22, [1], 1-7, 1955, 0031-8248, http://links.jstor.org/sici?sici=0031-8248%28195501%2922%3A1%3C1%3AARSA%3E2.0.CO%3B2-Q

33. (Boehm, Barry W., Port, D, Yang, Y, and Bhuta, J 2003)
Boehm, Barry W., Port, D, Yang, Y, and Bhuta, J, "Not All CBS Are Created Equally COTS-intensive Project Types", Proc. of the 2nd International Conference on COTS-Based Software Systems. Ottawa, Canada, February, 2003, LNCS, 2580, 36-50, 2003, Springer-Verlag, Berlin Heidelberg New York , http://springerlink.metapress.com/content/rv4d6wkewwtegq6w/?p=e9a9b9e2f0774eae972db7899f94c042&pi=0

34. Bonaccorsi, A., Giannangeli, S., & Rossi, C. 2006. Entry Strategies Under Competing Standards: Hybrid Business Models in the Open Source Software Industry. *Management Science*, 52(7): 1085-1098.

35. (Bradner, S 1996)
Bradner, S, "RFC: 2026 The Internet Standards Process", 1996, http://www.ietf.org/rfc/rfc2026.txt

36. (Brownsword, Lisa, Oberndorf, Tricia, and Sledge, Carol A. 2000)
Brownsword, Lisa, Oberndorf, Tricia, and Sledge, Carol A., "Developing New Processes for COTS Based Systems", IEEE Software, 17, [4], 48-55, 2000, http://dx.doi.org/10.1109/52.854068

37. (Bruce, Gary L. and Mottishaw, Peter 15-5-2005)
Bruce, Gary L. and Mottishaw, Peter, "Open OSS Catalyst White Paper", 15-5-2005, Telemanagement Forum, www.tmforum.org

38. (Bruce, Gary L., Robson, P., and Spaven, R. 2006)
Bruce, Gary L., Robson, P., and Spaven, R., "OSS opportunities in open source software - CRM and OSS standards", BT Technology Journal, 24, 127-140, 2006, Kluwer Academic Publishers, Hingham, MA, USA , http://dx.doi.org/10.1007/s10550-006-0028-9

39. (Bruce, Gary L., Wittgreffe, J. P., Potter, J. M., and Robson, P. 2005a)
Bruce, Gary L., Wittgreffe, J. P., Potter, J. M., and Robson, P., "The potential for open source software in telecommunications operational support systems", BT Technology Journal, 23, 79-95, 2005a, Kluwer Academic Publishers, Hingham, MA, USA , http://dx.doi.org/10.1007/s10550-005-0033-2

40. (Bruce, Gary L., Wolland, Ståle, Wiener, József, Silva, Nuno, and Ryan, Allan 2005b)
Bruce, Gary L., Wolland, Ståle, Wiener, József, Silva, Nuno, and Ryan, Allan, "Eurescom Study P1552 Report (D1): Open Source for Next Generation OSS - Issues and Challenges", 2005b, EURESCOM, Heidelberg, Germany ,

http://www.eurescom.de/activities/studyprogrammes/studyresults/P1500-series/P1552-D1.asp [Accessed Sept 2007]

41. (Brunelli, Mark 2005)
Brunelli, Mark, "Broadband networking firm gets hooked up with OpenNMS", SearchOpenSource.com, 2005, TechTarget, http://searchenterpriselinux.techtarget.com/originalContent/0,289142,sid39_gci1135868,00.html [Accessed Sept 2007]

42. (BT 2006)
BT, "BT Wholesale Web Call Connect Service and Interface Description, Suppliers Information Note, SIN 416, Issue 1.1, March 2006", 2006, Http://www1.btwebworld.com/sinet/418v1p1.pdf [Accessed Sept 2007]

43. (BT 2007a)
BT, "BT Technology Journal", 2007a, Springer, Netherlands 1573-1995, http://www.springerlink.com/content/1573-1995/

44. (BT 2007b)
BT, "BT Vision", 2007b, http://www.btvision.bt.com/ [Accessed Sept 2007]

45. (BT 2007c)
BT, "BT's 21st Century Network - Website", 2007c, http://www.btplc.com/21cn/ [Accessed Sept 2007]

46. (Bumpus, W. 2000)
Bumpus, W. 2000. *Common information model implementing the object model for enterprise management*. New York, N.Y: Wiley, ISBN.0471353426 (cloth : alk. paper) , http://www.loc.gov/catdir/toc/onix01/99089371.html

47. (Buschmann, Andreas and Hoeckelmann, Gerd 2007)
Buschmann, Andreas and Hoeckelmann, Gerd, "Trouble ticketing in action at Vodafone", Telemanamgent World, Nice, May 2007, Session SYS/SOFT23, 2007,

48. (Caruso, Francesco and Umar, Amjad 2004)
Caruso, Francesco and Umar, Amjad, "Architectures to Survive Technological and Business Turbulences", Information Systems Frontiers, 6, 9-21, 2004, Kluwer Academic Publishers, Hingham, MA, USA 1387-3326, http://dx.doi.org/10.1023/B:ISFI.0000015871.52897.03

49. (Casadesus-Masanell, Ramon and Ghemawat, Pankaj 1-7-2006)
Casadesus-Masanell, Ramon and Ghemawat, Pankaj, "Dynamic Mixed Duopoly: A Model Motivated by Linux vs. Windows", Management Science, 52, [7], 1072-1084, 1-7-2006, http://dx.doi.org/10.1287/mnsc.1060.0548

50. (Casey, Hugh R 2002)
Casey, Hugh R, "The Simple Event Monitor, A Tool for Network Management", Thesis submitted in partial completion of a Master of Science, 2002, University of Colorado, CO, USA , http://www.rickcasey.net/html_pages/thesis_on_web/thesis.pdf [Accessed Sept 2007]

51. (Castells, Manuel 2000)
Castells, Manuel, "The Information Age: Economy, Society and Culture. Volume I: The Rise of the Network Society, Volume II: The Power of Identity , Volume III: End of Millennium", 2000, Oxford University Press, Oxford

52. (Castor 2007)
Castor, "Castor XML Framework Project Homepage", 2007, http://www.castor.org/xml-framework.html [Accessed Sept 2007]

53. (CENELEC 2006)
CENELEC, "CENELEC Internal Regulations Part 3: Rules for the structure and drafting of CEN/CENELEC Publications (ISO/IEC Directives - Part 2, modified) 2006-12 corrected version", 2006, European Committee for Electrotechnical Standardization (CENELEC), http://www.cen.eu/boss/supporting/reference+documents/2006cencenelecirp3defcorrected.pdf [Accessed Sept 2007]

54. (Chapman, Chris and Ward, Steven 2003)
Chapman, Chris and Ward, Steven, "Project Risk Management", - Processes, Techniques and Insights (2nd Edition), 2003, John Wiley and Sons Ltd, Chichester

55. (Chapman, C. B. & Ward, S. 2002)
Chapman, C. B. & Ward, S. 2002. *Managing project risk and uncertainty - a constructively simple approach to decision making*. Chichester: Wiley, ISBN.ISBN 0470847905

56. (Chapman, Chris B. and Ward, Stephen 2004)
Chapman, Chris B. and Ward, Stephen, "Why risk efficiency is a key aspect of best practice projects", International Journal of Project Management, 22, [8], 619-632, 2004, Elsevier, http://dx.doi.org/10.1016/j.ijproman.2004.05.001

57. (Chapman, Chris B. and Ward, Stephen 2003)
Chapman, Chris B. and Ward, Stephen, "Project Risk Management - Processes, Techniques and Insights (2nd Edition)", 2003, John Wiley and Sons Ltd, Chichester

58. (Checkland, Peter and Howell, Sue 1998)
Checkland, Peter and Howell, Sue, "Action Research: Its Nature and Validity", Systemic Practice and Action Research, 11, [1], 9-21, 1998, Springer, Netherlands 1573-9295, http://dx.doi.org/10.1023/A:1022908820784

59. (Chein, Isidor, Cook, Stuart W, and Harding, John 2-1-1948)

Chein, Isidor, Cook, Stuart W, and Harding, John, "The field of action research", American Psychologist., 3, [2], 43-50, 2-1-1948, American Psychological Association 0003-066X, http://dx.doi.org/10.1037/h0053515

60. (Cheol-Seong, Kim, Joong-Goo, Song, Dong-Sik, Yoon, and Kyung-Sup, Sun 2004)
Cheol-Seong, Kim, Joong-Goo, Song, Dong-Sik, Yoon, and Kyung-Sup, Sun, "An NGOSS-based automatic service delivery system using EAI technology", Network Operations and Management Symposium, 2004.NOMS 2004.IEEE/IFIP, 2, 87-100, 2004, 1542-1201

61. (Cheol-Seong, Kim, Ok, Kisang, Hong, Daniel W, Chung, Byungsoo, and Bang, Yoon-Hak 2006)
Cheol-Seong, Kim, Ok, Kisang, Hong, Daniel W, Chung, Byungsoo, and Bang, Yoon-Hak, "NGOSS based convergent OSS framework toward business agility: KT case", International Journal of Network Management, 16, [6], 399-412, 2006, John Wiley & Sons, Inc, New York, NY, USA 1099-1190, http://dx.doi.org/10.1002/nem.629

62. (Christensen, Clayton M. 2003)
Christensen, Clayton M., "The Innovator's Dilemma", : the revolutionary book that will change the way you do business, 2003, New York : HarperBusiness Essentials, 2003A Harper business book, 0060521996

63. (Christensen, Clayton M. 16-3-2004)
Christensen, Clayton M., "Open Source: Capturing the Upside While Avoiding the Downside", (Audio Recording of Lecture), 16-3-2004, ITConversations, http://www.itconversations.com/shows/detail135.html [Accessed Sept 2007]

64. (Cisco 2007)
Cisco, "CISCO Model Wizard Project", 2007, http://sourceforge.net/projects/modelwizard/

65. (Coase, Ronald A. 1937)
Coase, Ronald A., "The Nature of the Firm", Economica, New Series, 4, [16], 386-405, 1937, http://links.jstor.org/sici?sici=0013-0427%28193711%292%3A4%3A16%3C386%3ATNOTF%3E2.0.CO%3B2-B

66. (Cooke, Chris 2006)
Cooke, Chris, "Knowledge-based Value and Intellectual Property", Workshop: Creation of Value in a Knowledge-Based Economy 5-6 April 2006, 2006, Univerisity of Lancaster, Lancaster, UK , http://www.opencapital.net/papers/Valueknowledge-based.pdf [Accessed Sept 2007]

67. (Coughlan, Paul and Coghlan, David 2-1-2002)
Coughlan, Paul and Coghlan, David, "Action research for operations management", International Journal of Operations & Production Management, 22, [2], 220-240, 2-1-2002, MCB UP Ltd, UK 0144-3577, http://dx.doi.org/10.1108/01443570210417515

68. (COVAD 2007)
COVAD, "COVAD ILEC-TT OSS/J Trouble Ticket Adaptor Homepage", 2007, COVAD Communications, https://ossj-ilectt.dev.java.net/ [Accessed Sept 2007]

69. (Crane, P. 2005)
Crane, P., "A new service infrastructure architecture.", BT Technology Journal, 23, [1], 15-27, 2005, Springer, Netherlands ISSN 1573-1995, http://dx.doi.org/10.1007/s10550-005-0104-4

70. (CruseControl 2007)
CruseControl, "CruiseControl continuous build system Project Homepage", 2007, http://cruisecontrol.sourceforge.net/ [Accessed Sept 2007]

71. (Cryer, Pat 2002)
Cryer, Pat, "The Research Student's Guide to Success, 2nd Edition", 2002, Open University Press, Buckingham

72. (CVS 2007)
CVS, "CVS - Concurrent Versions System Project Homepage", 2007, http://www.nongnu.org/cvs/ [Accessed Sept 2007]

73. (Darling, J. and Sauvage, A. 2005)
Darling, J. and Sauvage, A., "The application environment", BT Technology Journal, 23, [1], 82-89, 2005, Springer, Netherlands ISSN 1573-1995, http://dx.doi.org/10.1007/s10550-005-0109-z

74. (Davison, R. & Martinsons, M. G. 2007)
Davison, R. & Martinsons, M. G. 2007. Action Research and Consulting - Hellish Partnership or Heavenly Marriage. In N. F. Kock (Ed.), *Information systems action research: an applied view of emerging concepts and methods*: 377-394. New York, N.Y: Springer,.ISBN 9780387360591 (hbk.)

75. (Davison, Robert, Matinsons, Maris G, and Kock, Ned F. 1-1-2004)
Davison, Robert, Matinsons, Maris G, and Kock, Ned F., "Principles of Canonical Action Research", Information Systems Journal, 14, [1], 65-86, 1-1-2004, Blackwell Publishing, UK 1365-2575, http://dx.doi.org/10.1111/j.1365-2575.2004.00162.x

76. (de Bruijn, Brendan, Schickentanz, Regine, Grimm, Lothar, Wilhelm, Uwe, Chainho, Paulo, Vogel, Viet, and Mäkinen, Susanna 5-1-2001)
de Bruijn, Brendan, Schickentanz, Regine, Grimm, Lothar, Wilhelm, Uwe, Chainho, Paulo, Vogel, Viet, and Mäkinen, Susanna, "Eurescom Study P1044 Report (D1):Aspects of Open Source in Telecommunications : A strategic study on the use of Open Source in a telecommunications operator's environment", 5-1-2001, Eurescom, Heidelberg, Germany , http://www.eurescom.de/Public/projects/p1000-series/P1044/default.asp

77. (Dedrick, Jason and West, Joel 2004)
Dedrick, Jason and West, Joel, "An exploratory study into open source platform adoption", Proceedings of the 37th Annual Hawaii International Conference on System Sciences, 2004, 2004,

ISBN: 0-7695-2056-1, http://dx.doi.org/10.1109/HICSS.2004.1265633

78. (DiBona, C., Ockman, S., & Stone, M. 1999)
DiBona, C., Ockman, S., & Stone, M. 1999. *Open sources - voices from the open source revolution*. Beijing: O'Reilly, ISBN.1565925823 , http://www.oreilly.com/catalog/opensources/book/toc.html

79. (Dick, Bob 1993)
Dick, Bob, "Thesis resource paper : You want to do an action research thesis? - How to conduct and report action research", 1993, Southern Cross University, Australia , http://www.scu.edu.au/schools/gcm/ar/art/arthesis.html [Accessed Sept 2007]

80. (Dick, Bob 1997)
Dick, Bob, "Approaching an action ressearh thesis: an overview", 1997, Southern Cross University, Australia , http://www.scu.edu.au/schools/gcm/ar/arp/phd.html [Accessed Sept 2007]

81. (Dickerson, K. R 2004)
Dickerson, K. R, "Standards as an Enabler for Next Generation Networks", BT Technology Journal, 22, [2], 39-47, 2004, Springer, Netherlands ISSN 1573-1995, http://dx.doi.org/10.1023/B:BTTJ.0000033469.84056.9f

82. (Dinkelacker, Jamie, Garg Pankaj K., Miller, Rob, and Nelson, Dean 2002)
Dinkelacker, Jamie, Garg Pankaj K., Miller, Rob, and Nelson, Dean, "Progressive open source", Proceedings of the 24th International Conference on Software Engineering, (May 19-25, 2002, Orlando, Florida), 2002, ACM Press, New York, NY , http://doi.acm.org/10.1145/581339.581363

83. (Dixit, A. K. & Pindyck, R. S. 1994)
Dixit, A. K. & Pindyck, R. S. 1994. *Investment under uncertainty*. Princeton, N.J: Princeton University Press, ISBN.0691034109

84. (DMTF 2003)
DMTF, "DEN LDAP Mapping DMTF Technical Note", 2003, http://www.dmtf.org/education/technote_DENMapping.pdf [Accessed Sept 2007]

85. (DMTF 2004)
DMTF, "DMTF WBEMsource initiative: open source enterprise management RI's.", 2004, http://www.wbemsource.org/ [Accessed Sept 2007]

86. (DMTF 2005)
DMTF, "Distributed Management Task force (DMTF) Web-Based Enterprise Management (WBEM) Initiative", 2005, http://www.dmtf.org/standards/wbem [Accessed Sept 2007]

87. (DMTF 2006)
DMTF, "Common Information Model (CIM) Standards", 2006, Distributed Management Task Force (DMTF), http://www.dmtf.org/standards/cim/ [Accessed Sept 2007]

88. (DMTF 2007a)
DMTF, "Distributed Management Task Force Inc.", Address: 1001 SW 5th Avenue, #1100 Portland, OR 97204, 2007a, http://www.dmtf.org [Accessed Sept 2007]

89. (DMTF 2007b)
DMTF, "DMTF MOF: Managed Object Format", 2007b, http://www.dmtf.org/education/mof/ [Accessed Sept 2007]

90. (Docbook 2007)
Docbook, "Docbook Project Homepage", 2007, http://docbook.sourceforge.net/ [Accessed Sept 2007]

91. (Donnelly, W. and Wade, Vincent 2001)
Donnelly, W. and Wade, Vincent, "Reusable Building Blocks for Telecommunications Communication Integration", Third international workshop on Design of Reliable Communication Networks (DRCN), Budapest, 2001, IEEE, http://emerge.tmit.bme.hu/PAPERS/88.PDF [Accessed Sept 2007]

92. (Driver, Mark 2007)
Driver, Mark, "Gartner: Ignoring Open Source Not An Option", Urlocker, Zack (Eds.), Report on Gartner Open Source Summit, September 2007, infoworld, 2007, http://weblog.infoworld.com/openresource/archives/2007/09/gartner_ignorin.html

93. (Duke, A, Davies, J., and Richardson, M. 2005)
Duke, A, Davies, J., and Richardson, M., "Enabling a scalable service-oriented architecture with semantic Web Services.", BT Technology Journal, 23, [3], 191-201, 2005, Springer, Netherlands ISSN 1573-1995, http://dx.doi.org/10.1007/s10550-005-0041-2

94. (Duke, A., Richardson, M., Watkins, S., Wahler, A., and Schreder, B 2006)
Duke, A., Richardson, M., Watkins, S., Wahler, A., and Schreder, B, "Service assurance across organisational boundaries with semantic mediation", BT Technology Journal, 24, [1], 153-160, 2006, Springer, Netherlands ISSN 1573-1995, http://dx.doi.org/10.1007/s10550-006-0030-0

95. (DVB-H 2007)
DVB-H, "DVB-H, the specification for bringing broadcast services to battery-powered handheld receivers", 2007, http://www.dvb-h.org/ [Accessed Sept 2007]

96. (EasyMock 2007)
EasyMock, "EasyMock Project Homepage", (EasyMock provides Mock Objects for interfaces in JUnit tests), 2007, [Accessed Sept 2007] , http://www.easymock.org/

97. (ebXML 2007)
ebXML, "Electronic Business using eXtensible Markup Language (ebXML) Homepage", 2007,

http://www.ebxml.org/ [Accessed Sept 2007]

98. (Eclipse 2007a)
Eclipse, "Eclipse IDE Project Homepage", 2007a, Eclipse Foundation, http://www.eclipse.org/ [Accessed Sept 2007]

99. (Eclipse 2007b)
Eclipse, "EMF - Eclipse Modelling Framework Project Homepage", 2007b, Eclipse Foundation, http://www.eclipse.org/modeling/emf/ [Accessed Sept 2007]

100. (Economides, Nicholas and Flyer, Patrick 1998)
Economides, Nicholas and Flyer, Patrick, "Equilibrium Coalition Structures in Markets for Network Goods", ANNALES D' 'ECONOMIE ET DE STATISTIQUE, N 49/50, 1998, http://www.pse.ens.fr/adres/anciens/n4950/vol4950-14.pdf [Accessed Sept 2007]

101. (Economides, Nicholas and Skrzypacz, Andrzej 2003)
Economides, Nicholas and Skrzypacz, Andrzej, "Standards Coalitions Formation and Market Structure in Network Industries (Working Paper no. EC-03-08)", 2003, Stern School of Business, N.Y.U, New York, NY, USA , http://www.stern.nyu.edu/networks/Standards.pdf [Accessed Sept 2007]

102. (Economides, Nicolas 1996)
Economides, Nicolas, "The Economics of Networks", International Journal of Industrial Organization, 14, [6], 673-699, 1996, Elsevier Science B.V., http://dx.doi.org/10.1016/0167-7187(96)01015-6

103. (Edwards, Silvia Lauretta and Bruce Christina 10-1-2002)
Edwards, Silvia Lauretta and Bruce Christina, "Reflective Internet searching: an action research model", The Learning Organization, 9, [4], 180-188, 10-1-2002, MCB UP Ltd 0969-6474, http://dx.doi.org/10.1108/09696470210428903

104. (Elgin, Ben 2005)
Elgin, Ben, "Managing Google's Idea Factory", Business Week, 2005, McGraw-hill, http://www.businessweek.com/magazine/content/05_40/b3953093.htm [Accessed Sept 2007]

105. (EPSRC 2006)
EPSRC, "The EPSRC Engineering Doctorate Scheme Good Practice Guidance", (Previously published as 'The EPSRC Engineering Doctorate Scheme Best Practice for Centres, Sponsoring Organisations, Supervisors and Research Engineers', 2002), 2006, http://www.epsrc.ac.uk/PostgraduateTraining/EngineeringDoctorates/EngDGoodPracticeGuidance.htm [Accessed Sept 2007]

106. (Esperanza, Marcos 2005)
Esperanza, Marcos, "Software engineering research versus software development", SIGSOFT Softw.Eng.Notes, 30, 1-7, 2005, ACM Press, New York, NY, USA 0163-5948

107. (Estay-Niculcar, C. A. and Pastor-Collado, J. A. 2002)
Estay-Niculcar, C. A. and Pastor-Collado, J. A., "A Maturity Model For Information Systems Action-research Project Management", Proceedings of the Tenth European Conference on Information Systems (Wrycza S ed.), 2002, Gdansk University, Gdansk Poland 83-7326-077-3, http://is2.lse.ac.uk/asp/aspecis/20020008.pdf [Accessed Sept 2007]

108. (ETSI 1998a)
ETSI, "ETS 300 421 V1.1.2 (1997-08) Digital broadcasting systems for television, sound and data services; Framing structure, channel coding and modulation for 11/12 GHz satellite services (DVB-S)", 1998a,

109. (ETSI 1998b)
ETSI, "ETSI EN 300 814 Digital Video Broadcasting (DVB);DVB interfaces to Synchronous Digital Hierarchy (SDH) networks", 1998b, ETSI

110. (ETSI 2004)
ETSI, "ETSI EN 300 744 V1.5.1 (2004-06) European Standard (Telecommunications series)Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television", 2004, ETSI

111. (ETSI 2006a)
ETSI, "EN 300 401 V1.4.1 (2006-06) Technical Specification: Radio broadcasting systems; Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers", 2006a, ETSI

112. (ETSI 2006b)
ETSI, "ETSI EN 300 401 Radio Broadcasting Systems; Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers", 2006b, European Telecommunications Standards Institute

113. (ETSI 2006c)
ETSI, "ETSI TR 101 891 V1.1.1 (2001-02) Technical Report. Digital Video Broadcasting (DVB); Professional Interfaces: Guidelines for the implementation and usage of the DVB Asynchronous Serial Interface (ASI)", 2006c, ETSI

114. (ETSI 2007)
ETSI, "ETSI TS 102 032 V1.1.1 (2002-04) Technical Specification Digital Video Broadcasting (DVB); SNMP MIB for test and measurement applications in DVB systems", 2007, ETSI

115. (Farragher, Edward J., Kleiman, Robert T., and Sahu, Anandi P. 1999a)
Farragher, Edward J., Kleiman, Robert T., and Sahu, Anandi P., The Engineering Economist, 44, [2], 137-150, 1999a, Taylor & Francis, http://dx.doi.org/10.1080/00137919908967513

116. (Farragher, Edward J., Kleiman, Robert T., and Sahu, Anandi P. 1999b)

Farragher, Edward J., Kleiman, Robert T., and Sahu, Anandi P., The Engineering Economist, 44, [2], 137-150, 1999b, Taylor & Francis, http://dx.doi.org/10.1080/00137919908967513

117. (Feller, Joseph and Fitzgerald, Brian 2000)
Feller, Joseph and Fitzgerald, Brian, "A framework analysis of the open source software development paradigm", Proceedings of the twenty first International Conference on Information Systems (ICIS '00), December 10-13, 2000, Brisbane, Queensland, Australia, 58-69, 2000, ISBN:ICIS2000-X, http://portal.acm.org/citation.cfm?id=359640.359723#

118. (Fielding, Thomas 2000)
Fielding, Thomas, "Representational State Transfer (REST)", (Chapter 5), Architectural Styles and the Design of Network-based Software Architectures (PhD Dissertation), 2000, University of California, Irvine, CA, USA , http://roy.gbiv.com/pubs/dissertation/top.htm [Accessed Sept 2007]

119. (Fink, M. 2002)
Fink, M. 2002. *The business and economics of Linux and Open Source*. Upper Saddle River, N.J., London : Prentice Hall PTR, ISBN.0130476773

120. (Fitzgerald, Brian and Kenny, Tony. 2003)
Fitzgerald, Brian and Kenny, Tony., "Open source software in the trenches: Lessons from a large scale implementation", Proceedings of 24th International Conference on Information Systems (ICIS) (Seattle, Washington, Dec.14- 17, 2003, 2003, http://www.lse.ac.uk/collections/informationSystems/newsAndEvents/2004events/openSourceSoftwareInTheTrenches.htm [Accessed Sept 2007]

121. (Fitzgerald, Brian and Kenny, Tony. 2004)
Fitzgerald, Brian and Kenny, Tony., "Developing an Information Systems Infrastructure with Open Source Software", IEEE Software, 21, [1], 50-55, 2004, http://dx.doi.org/10.1109/MS.2004.1259216

122. (FlowThrough 2000a)
FlowThrough, "FlowThrough Project Home Pages", 2000a, http://www.cs.ucl.ac.uk/research/flowthru/ see also http://cordis.europa.eu/infowin/acts/rus/projects/ac335.htm [Accessed Sept 2007]

123. (FlowThrough 21-1-2000b)
FlowThrough, "Implementing Integrated Management Business Processes using Reusable Components - A Report on FlowThru Project. Editor David Lewis", 21-1-2000b, FLOWTHRU Consortium, http://www.cs.ucl.ac.uk/research/flowthru/content/finalrpt/flowthru-rpt.pdf [Accessed Sept 2007]

124. (Fogel, K. & Bar, M. 2001)
Fogel, K. & Bar, M. 2001. *Open source development with CVS*. Scottsdale, AZ : Coriolis Group Books, ISBN.158880173X

125. (FOKUS 2007)
FOKUS, "The FOKUS Open Source IMS Core (OSIMS) Project", 2007, http://www.fokus.fraunhofer.de/bereichsseiten/testbeds/ims_playground/index.php?lang=en [Accessed Sept 2007]

126. (FORM 2002)
FORM, "FORM - Federated ORganisations Management, EU IST 5th framework project contract IST-1999-103571", Form Home page: 2002, http://www.cs.ucl.ac.uk/research/form/web/ See also http://cordis.europa.eu/fetch?CALLER=PROJ_IST&ACTION=D&DOC=94&CAT=PROJ&QUERY=1176496220984&RCN=57185 [Accessed Sept 2007]

127. (Fowler, Martin 2006)
Fowler, Martin, "Continuous Integration 01 May 2006", 2006, http://www.martinfowler.com/articles/continuousIntegration.html [Accessed Sept 2007]

128. (Fowler, M. 2004)
Fowler, M. 2004. *UML distilled a brief guide to the standard object modeling language*. Boston: Addison-Wesley, ISBN.0321193687 (pbk. : alk. paper)

129. (Fowler, Martin 2003)
Fowler, Martin, "Platform Independent Malapropism", 2003, http://martinfowler.com/bliki/PlatformIndependentMalapropism.html [Accessed Sept 2007]

130. (Fransman, Martin 2001)
Fransman, Martin, "Evolution Of The Telecommunications Industry Into The Internet Age (to appear in World Telecommunications Markets: The International Handbook of Telecommunications Economics, Volume III, Edward Elgar Publishing, forthcoming)", 2001, TelecomVisions.com, http://telecomvisions.com/articles/pdf/FransmanTelecomsHistory.pdf [Accessed Sept 2007]

131. (Fransman, Martin 2002a)
Fransman, Martin, "Mapping the evolving telecoms industry: the uses and shortcomings of the layer model", Telecommunications Policy, 26, [9-10], 473-483, 2002a, http://dx.doi.org/10.1016/S0308-5961(02)00027-7

132. (Fransman, M. 2002b)
Fransman, M. 2002b. *Telecoms in the Internet age from boom to bust to--?* Oxford: Oxford University Press, ISBN.0199257000

133. (Freeview 2007)
Freeview, "Freeview, DTV Services Ltd", 2007, http://www.freeview.co.uk [Accessed Sept 2007]

134. (FT Online 2006)

FT Online, "Google to beat TV in race for ad revenues November 2 2006", 2006, Financial Times Online, http://www.ft.com/cms/s/91d89384-6978-11db-952e-0000779e2340.html [Accessed Sept 2007]

135. (Gacek, Cristina 2004)
Gacek, Cristina, "Standards and Open Source Software: Twins, Cousins, or Just Neighbours?, Technical Report CS-TR:867", 2004, School of Computing Science, Newcastle University, Newcastle, England , http://roguc.ncl.ac.uk/deposit_details.php?deposit_id=256 [Accessed Sept 2007]

136. (Gacek, Cristina, Lawrie, Tony, and Arief, Budi 2001)
Gacek, Cristina, Lawrie, Tony, and Arief, Budi, "The Many Meanings of Open Source, Technical Report CS-TR: 737", 2001, Department of Computing Science, Newcastle University, Newcastle , http://www.cs.ncl.ac.uk/research/pubs/trs/papers/737.pdf [Accessed Sept 2007]

137. (Galbraith, R. I. 2002)
Galbraith, R. I., "Standards for Broadband Customer Premises Equipment", BT Technology Journal, 20, [2], 82-95, 2002, Springer, Netherlands ISSN 1573-1995, http://dx.doi.org/10.1023/A:1015644523015

138. (Gallen C.R. 1-5-2003a)
Gallen C.R., "Engineering Doctorate (Eng D) Enterprise Management Systems for the Telecommunications Industry Program Plan ENG001", [13.0], 1-5-2003a,

139. (Gallen C.R. 19-9-2003b)
Gallen C.R., "Engineering Doctorate (Eng D) Enterprise Management Systems for the Telecommunications Industry Technology Review ENGD002", 19-9-2003b,

140. (Gallen C.R. 5-11-2003c)
Gallen C.R., "Invocom Eng D Strategy Analysis II (Power Point presentation strategy analysis II;2 5-11-03.ppt)", 5-11-2003c,

141. (Gallen C.R. 17-10-2003d)
Gallen C.R., "Invocom EngD Strategy Thinking - Powerpoint Presentation (Invocom Strategy - Eng D 17-10-03.ppt)", 17-10-2003d,

142. (Gallen, Craig R. 2003b)
Gallen, Craig R., "Engineering Doctorate (Eng D) Enterprise Management Systems for the Telecommunications Industry Technology Review, Internal Report (C Gallen Technology review;16 19-9-03.doc)", 2003b,

143. (Gallen, Craig R. 1-5-2003a)
Gallen, Craig R., "Engineering Doctorate (Eng D) Enterprise Management Systems for the Telecommunications Industry Program Plan ENG001", [13.0], 1-5-2003a,

144. (Gallen, Craig R. and Carson, Doug 2005a)
Gallen, Craig R. and Carson, Doug, "Using JBoss and the Open Source approach in the OpenOSS project", Jboss World Telecoms Track Presentation, 2005a, Barcelona , http://www.jboss.com/pdf/jbw_barc/telcom/openOSS_project.pdf [Accessed Sept 2007]

145. (Gallen, Craig R. and Pearson, S 18-1-2005)
Gallen, Craig R. and Pearson, S, "The future of open source in Operational Support Systems", Conference on Strategic OSS for next generation operators, 18-1-2005, IPQC, London

146. (Gallen, Craig R. and Reeve, Jeff S. 2005b)
Gallen, Craig R. and Reeve, Jeff S., "Investigating the Feasibility of Open Development of Operations Support Solutions", 689-702, 2005b, IEEE, Piscataway, NJ 2005 9th IFIP/IEEE International Symposium on Integrated Network Management, 15-5-2005b, http://dx.doi.org/10.1109/INM.2005.1440842

147. Gallen, C. R. & Reeve, J. S. 2005a. Enterprise Management Systems for the Telecommunications Industry. University of Southampton.

148. (Gallen, Craig R. and Reeve, Jeff S. 2005c)
Gallen, Craig R. and Reeve, Jeff S., "The OpenOSS Project - An open Source operations Support Lab for the Telecommunications Industry", Proceedings of Workshop on Free/Open Source Software, Technologies and Content (FOSTEC 05), The 9th World Multiconference on Systems, Cybernetics and Informatics.(IX), Orlando, FA, Volume 1X, 2005c, Orlando , http://eprints.ecs.soton.ac.uk/12510/

149. (Gallen, Craig R. and Reeve, Jeff S. 2006)
Gallen, Craig R. and Reeve, Jeff S., "Using Open Source to Realise an NGOSS Proof of Concept", 2006 IEEE/IFIP Network Operations and Management Symposium - Application Sessions - NOMS 2006, April 3-7, 2006, 2006, http://dx.doi.org/10.1109/NOMS.2006.1687603

150. (Gehring, Robert A. 2006)
Gehring, Robert A., "The institutionalization of Open Source", Poiesis & Praxis: International Journal of Technology Assessment and Ethics of Science, 4, [1], 54-73, 2006, Springer, Berlin / Heidleberg 1615-6617, http://dx.doi.org/10.1007/s10202-005-0012-1

151. (Georgalas, N and Azmoodeh, M. 2004)
Georgalas, N and Azmoodeh, M., "Using MDA in Technology-independent Specifications of NGOSS Architectures", Proceedings of the First European Workshop of Model Driven Architecture with emphasis on Industrial Applications (MDA-IA 2004), Enschede, The Netherlands, 17-18 March 2004 (CTIT Technical Report TR-CTIT-04-12), 2004, University of Twente 1381-3625, http://www.dcs.bbk.ac.uk/~dgcor02/MDA-IA2004/MDA-IA2004.htm [Accessed Sept 2007]

152. (Georgalas, N and Bagley, C 2005)
Georgalas, N and Bagley, C, "Using policies in highly configurable component-based NGOSS", BT Technology Journal, 23, [3], 149-161, 2005, Springer, Netherlands ISSN 1573-1995, http://dx.doi.org/10.1007/s10550-005-0038-x

153. (German, D. M. 2005)
German, D. M. 2005. Software Engineering Practices in the GNOME Project. In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*: 212-225. Cambridge, Mass: MIT Press,.ISBN 0-262-06246-1

154. (Ghosh, Rashab Aiyer 2005a)
Ghosh, Rashab Aiyer, "An Economic Basis for Open Standards (Deliverable D4)", FLOSSPOLS Project - Free/Libre/Open Source Software Policy Support, 2005a, UNU-MERIT United Nations University, Maastricht, NL , http://flosspols.org/deliverables/FLOSSPOLS-D04-openstandards-v6.pdf [Accessed Sept 2007]

155. (Ghosh, R. A. 2005b)
Ghosh, R. A. 2005b. Understanding Free Software Developers Findings from the FLOSS Study. In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*: 23-45. Cambridge, Mass: MIT Press,.ISBN 0-262-06246-1

156. (Giacomo, P. D. 2005)
Giacomo, P. D., "COTS and Open Source Software Components: Are They Really Different on the Battlefield?", Proc. of the 4th International Conference on COTS-Based Software Systems. Bilbo, Spain, February, 2005, LNCS, 3412/2005, 301-310, 2005, Springer-Verlag, Berlin, Heidelberg, New York 978-3-540-24548-3, http://dx.doi.org/10.1007/b105900

157. (Giles, Joshua 2003)
Giles, Joshua, "Integrating OpenNMS with OpenManage Server Agent to Manage Dell PowerEdge Servers Running Linux", Dell Power Solutions, 2003, Dell, http://www.dell.com/content/topics/global.aspx/power/en/ps1q03_giles [Accessed Sept 2007]

158. (Giles, Steve 2000)
Giles, Steve, "Open Enterprise Management Software", 2000, Front Range UNIX Users Group (FRUUG), http://www.fruug.org/Archive/2000-04b/opennms.pdf [Accessed Sept 2007]

159. (Gladwell, M. 2000)
Gladwell, M. 2000. *The tipping point how little things can make a big difference*. Boston: Little, Brown, ISBN.0316316962

160. (Glass, Robert L., Ramesh, V., and Vessey, Iris 2004)
Glass, Robert L., Ramesh, V., and Vessey, Iris, "An analysis of research in computing disciplines", Commun.ACM, 47, [6], 89-94, 2004, ACM Press, New York, NY, USA 0001-0782, http://doi.acm.org/10.1145/990680.990686

161. (Goldman, Larry 2004a)
Goldman, Larry, "TMW Asia 2004: OSS Market Overview - An Asian Perspective", 2004a, OSS Observer, http://www.tmforum.org/browse.asp?catID=2103&sNode=2103&Exp=Y&linkID=29555

162. (Goldman, Larry 2004b)
Goldman, Larry, "TMW Nice 2004: The Global OSS Landscape", OSS Observer, 2004b, http://www.ossobserver.com/library/library12.pdf

163. (Google 2007)
Google, "Google Web Toolkit (GWT) Homepage", 2007, Google, http://code.google.com/webtoolkit/ [Accessed Sept 2007]

164. (Greenfield, Jack, Short, Kieth, Cook, Steve, Kent, Stuart, and Crupti, John 2004)
Greenfield, Jack, Short, Kieth, Cook, Steve, Kent, Stuart, and Crupti, John, "Software Factories: Assembling Applications with Patterns, Models, Frameworks, and Tools", 2004, Wiley, Indianapolis

165. (Griffin, Abbie and Hauser, John R. 1996)
Griffin, Abbie and Hauser, John R., "Integrating R&D and Marketing: A Review and Analysis of the Literature", Journal of Product Innovation Management, 13, [3], 191-215, 1996, Elsevier 0737-6782, http://dx.doi.org/10.1111/1540-5885.1330191

166. (Guardian 2002)
Guardian, "WorldCom goes bankrupt (Monday 22nd July 2002)", 2002, Guardian News and Media Limited, Manchester, UK , http://www.guardian.co.uk/worldcom/story/0,,759854,00.html [Accessed Sept 2007]

167. (Guba, E. G. & Lincoln, Y. S. 1989)
Guba, E. G. & Lincoln, Y. S. 1989. *Fourth generation evaluation*. Newbury Park, Calif: Sage Publications, ISBN.0803932359

168. (Gulliksen, Jan and Göransson, Bengt 2000)
Gulliksen, Jan and Göransson, Bengt, "Action research versus experimental methodologies for HCI: Report number: CID-118: Position paper for the NordiCHI workshop on "Quality and validity in HCI research"", 2000, Centre for User Oriented IT Design, Royal Institute of Technology, Stockholm, Sweden 1403-0721, http://citeseer.ist.psu.edu/559438.html [Accessed Sept 2007]

169. (Gurbani, Vijay K., Garvert, Anita, and Herbsleb, James D. 2006)
Gurbani, Vijay K., Garvert, Anita, and Herbsleb, James D., "A case study of a corporate open source development model", Proceeding of the 28th international conference on Software engineering, Shanghai, China, 2006, ACM Press, New York, NY, USA 1-59593-375-1,

http://doi.acm.org/10.1145/1134285.1134352

170. (Guth, Robert A 1-10-2004)
Guth, Robert A, "Microsoft Could Lose 1996 Patent That May Hinder Linux Software", The Wall Street Journal, Section B4, 1-10-2004, http://online.wsj.com/article/SB109658046156032952.html [Accessed Sept 2007]

171. (Hanson, R. & Tacy, A. 2007)
Hanson, R. & Tacy, A. 2007. *GWT in action : easy Ajax with the Google Web toolkit*. Greenwich, CT: Manning, ISBN.9781933988238

172. (Hayler, K. 1999)
Hayler, K., "UK DVB-T transmission; the challenges and solutions to achieve launch", IEE Colloquium on Digital Television - Where Is It and Where Is It Going? 16 March 1999 , (Ref.No.1999/072), 7-1-7/8, 1999, IEE, London , http://ieeexplore.iece.org/search/srchabstract.jsp?arnumber=773223&isnumber=16766&punumber=6290&k2dockey=773223@ieeecnfs [Accessed Sept 2007]

173. (Hazzan, O. 9-1-2002)
Hazzan, O., "The reflective practitioner perspective in software engineering education", Journal of Systems and Software, 63, [3], 161-171, 9-1-2002, Elsevier Inc, Amsterdam 0164-1212, http://dx.doi.org/10.1016/S0164-1212(02)00012-2

174. (Hazzan, O. and Tomayko, J. E. 2003)
Hazzan, O. and Tomayko, J. E., "The Reflective Practitioner Perspective in eXtreme Programming", Proceedings - Extreme Programming and Agile Methods - XP/Agile Universe 2003, LNCS, 2753, 51-61, 2003, Springer, Berlin 978-3-540-40662-4

175. (Hazzan, O. and Tomayko, J. E. 2005)
Hazzan, O. and Tomayko, J. E., "Reflection and abstraction in learning software engineering's human aspects", Computer, 38, [6], 39-45, 2005, 0018-9162, http://dx.doi.org/10.1109/MC.2005.200

176. (Henning, Michi 6-1-2006)
Henning, Michi, "The Rise and Fall of CORBA", ACM Queue, 4, [5], 6-1-2006, http://acmqueue.com/modules.php?name=Content&pa=showpage&pid=396 [Accessed Sept 2007]

177. (Henning, M. & Vinoski, S. 1999)
Henning, M. & Vinoski, S. 1999. *Advanced CORBA programming with C++*. Reading, Mass: Addison-Wesley, ISBN.0201379279 (pbk. : alk. paper)

178. (Hibbert, C. 1999)
Hibbert, C., "Issues in the implementation of a digital regional terrestrial service", IEE Colloquium on Digital Television - Where Is It and Where Is It Going? 16 March 1999 , (Ref.No.1999/072), 1-1/8, 1999, IEE, London , http://ieeexplore.ieee.org/iel5/6290/16766/00773217.pdf?isnumber=16766&prod=STD&arnumber=773217&arnumber=773217&arSt=1%2F1&ared=1%2F8&arAuthor=Hibbert%2C+C.

179. (Hibernate 2007)
Hibernate, "Hibernate Project Homepage", 2007, http://www.hibernate.org/

180. (Hirvensalo, Antero, Evokari, Juha, Feller, Jan, Pekkola, Kimmo, Turunen, Pekka, and Smeds, Riitta 2003)
Hirvensalo, Antero, Evokari, Juha, Feller, Jan, Pekkola, Kimmo, Turunen, Pekka, and Smeds, Riitta, "R&DNet Final Report. Process Innovation in Collaborative R&D. Lessons Learnt from the Telecommunications Industry", 2003, Helsinki University of Technology SimLab, Helsinki ISBN: 951-22-6494-3, http://www.simlab.tkk.fi/publications/RDNetFR.pdf [Accessed Sept 2007]

181. (Hoag, J. C. 2006)
Hoag, J. C., "An Architecture for Network Operations and Management Based on State and Services", Proceedings of The 1st International Workshop on Broadband Convergence Networks, 2006.(BcN 2006), 1-4, 2006, ISBN: 1-4244-0146-1, http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1662277

182. (Hobbs, C. 2004)
Hobbs, C. 2004. *A practical approach to WBEM/CIM management*. Boca Raton, FL: Auerbach, ISBN.0849323061 (alk. paper)

183. (Hobbs, Martin and Kier, Don 2005a)
Hobbs, Martin and Kier, Don, "Sidonis Statewise User Guide", 2005a, Sidonis, Bath, UK , http://www.sidonis.com/

184. (Hobbs, Martin and Kier, Don 2005b)
Hobbs, Martin and Kier, Don, "Sidonis StateWise User Guide [Pre Publication Version Lent by Authors]", 2005b, Sidonis, Bath, UK , http://www.sidonis.com/

185. (Holz, Hilary J., Applin, Anne, Haberman, Bruria, Joyce, Donald, Purchase, Helen, and Reed, Catherine 2006)
Holz, Hilary J., Applin, Anne, Haberman, Bruria, Joyce, Donald, Purchase, Helen, and Reed, Catherine, "Research methods in computing: what are they, and how should we teach them?", ITiCSE-WGR '06: Working group reports on ITiCSE on Innovation and technology in computer science education, 96-114, 2006, ACM Press, New York, NY, USA 1-59593-603-3, http://doi.acm.org/10.1145/1189215.1189180

186. (Hopkins, Jon 2000)
Hopkins, Jon, "Component primer", Communications of the ACM, 43, [10], 27-30, 2000, http://doi.acm.org/10.1145/352183.352198

187. (Hoppenbrouwers, S. J. B. A, Lindeman, Leonie, and Proper, H. A 2006)

Hoppenbrouwers, S. J. B. A, Lindeman, Leonie, and Proper, H. A, "Capturing Modeling Processes - Towards the MoDial Modeling Laboratory", On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops, Edited by: R. Meersman, Z. Tari, and P. Herrero, Lecture Notes in Computer Science, 4278/2006, 1242-1252, 2006, ISSN 0302-9743, http:/dx.doi.org/10.1007/11915072_27

188. (Hoppenbrouwers, S. J. B. A, Proper, H. A, and van der Weilde, Th. P. 2005a)
Hoppenbrouwers, S. J. B. A, Proper, H. A, and van der Weilde, Th. P., "Fundamental Understanding of the Act of Modelling. Technical report: ICIS-R05006", 2005a, Radboud University, Nijmegen , http://www.cs.ru.nl/icis/Research/reports/full/NIII-R05006.pdf [Accessed Sept 2007]

189. (Hoppenbrouwers, S. J. B. A, Proper, H. A, and van der Weilde, Th. P. 2005b)
Hoppenbrouwers, S. J. B. A, Proper, H. A, and van der Weilde, Th. P., "Understanding the Requirements on Modelling Techniques", Advanced Information Systems Engineering, Proceedings of 17th International Conference, CAiSE 2005, Porto, Portugal, June 13-17, 2005, 3520/2005, 2005b, Springer Verlag, NL 978-3-540-26095-0, http://dx.doi.org/10.1007/11431855_19

190. (Howell, S. 2004)
Howell, S. 2004. Themes, Iteration and Recoverability in Action Research. In B. Kapln, D. P. Truex, D. Wastel, T. Wood-Harper, & DeGross J.I. (Eds.), *Information Systems Research: Relevant Theory and Informed Practice*: Boston, USA: Kluwer Academic Publishers,.ISBN 1-4020-8094-8, http://oro.open.ac.uk/159/01/IFIP_8_2_AR.pdf

191. (Howlett, T. 2005)
Howlett, T. 2005. *Open source security tools practical applications for security*. Upper Saddle River, NJ: Prentice Hall, ISBN.0321194438 (pbk. : alk. paper) , http://www.loc.gov/catdir/toc/ecip0417/2004009479.html

192. (HP 2007a)
HP, "Hewlett Packard TeMIP Framework", 2007a, http://h20229.www2.hp.com/products/tmpfw/index.html [Accessed Sept 2007]

193. (HP 2007b)
HP, "HP OpenView", 2007b, http://www.openview.hp.com/ [Accessed Sept 2007]

194. (IBM 2007a)
IBM, "Free IBM sales/marketing help if using IBM free middleware", 2007a, http://www-304.ibm.com/jct09002c/isv/opensource/ [Accessed Sept 2007]

195. (IBM 2007b)
IBM, "IBM Tivoli Netcool Product Homepage", 2007b, IBM, http://www-306.ibm.com/software/tivoli/welcome/micromuse/ [Accessed Sept 2007]

196. (IEEE/IFIP 2007)
IEEE/IFIP, "IEEE/IFIP Network Operations & Management Symposium (NOMS)", 2007, http://www.noms.org [Accessed Sept 2007]

197. (IETF 1989)
IETF, "RFC 1095 CMOT - CMip Over Tcp/ip", 1989, http://www.ietf.org/rfc/rfc1095 [Accessed Sept 2007]

198. (IETF 1990)
IETF, "RFC 1157 A Simple Network Management Protocol (SNMP)", 1990, http://www.ietf.org/rfc/rfc1157

199. (IETF 2007)
IETF, "The Internet Engineering Task Force Homepage", 2007, http://www.ietf.org/ [Accessed Sept 2007]

200. (IFIP/IEEE 2007)
IFIP/IEEE, "IFIP/IEEE Symposium on Integrated Management", 2007, http://www.icce-im.org [Accessed Sept 2007]

201. (ILOG 2007)
ILOG, "ILOG Jrules Product Homepage", 2007, http://www.ilog.com/products/jrules/ [Accessed Sept 2007]

202. (Information Technology & People 2001)
Information Technology & People 2001. Special Issue on Information Systems Action Research. In F. Lau & N. F. Kock (Eds.), *Information,Technology & People*.ISBN 0959-3845, http://www.emeraldinsight.com/Insight/viewContainer.do?containerType=Issue&containerId=20054

203. (Invocom 2007)
Invocom, "Invocom ltd, 400 Thames Valley Park Drive, Thames Valley Park, Reading, Berkshire, England, RG6 1PT, Telephone: +44 (0) 118 965 3422, Email: info@invocom.com, Registered in England and Wales, number 03862165", 2007, http://www.invocom.com/ [Accessed Sept 2007]

204. (IPTEL 2007a)
IPTEL, "SIP Express Media Server (SEMS) Homepage", 2007a, http://developer.berlios.de/projects/sems/ [Accessed Sept 2007]

205. (IPTEL 2007b)
IPTEL, "SIP Express Router (SER) Homepage", 2007b, www.iptel.org/ser/ [Accessed Sept 2007]

206. (Irwin, Rodger 1998)

Irwin, Rodger, "What is FUD?", 1998, http://www.cavcomp.demon.co.uk/halloween/fuddef.html [Accessed Sept 2007]

207. (ITIL 2007)
ITIL, "IT Infrastructure Library (ITIL)", 2007, Office of Government Commerce (OGC), HM Treasury, London, UK , http://www.itil.co.uk/ [Accessed Sept 2007]

208. (ITU-T 1992a)
ITU-T, "ITU-T Recommendation X.700 Management framework for Open Systems Interconnection (OSI) for CCITT applications", 1992a, International Telecommunications Union ITU-T, Geneva , http://www.itu.int/rec/T-REC-X.700/en [Accessed Sept 2007]

209. (ITU-T 1992b)
ITU-T, "ITU-T Recommendation X.722 Information technology - Open Systems Interconnection - Structure of management information: Guidelines for the definition of managed objects", 1992b, International Telecommunications Union ITU-T, Geneva , http://www.itu.int/rec/T-REC-X.722/en [Accessed Sept 2007]

210. (ITU-T 1992c)
ITU-T, "ITU-T Recommendation X.733 Information Technology - Open Systems Interconnection - Systems Management: Alarm Reporting Function", 1992c, International Telecommunications Union ITU-T, Geneva , http://www.itu.int/rec/T-REC-X.733/en

211. (ITU-T 1997a)
ITU-T, "ITU-T Recommendation X.711 Information technology - Open Systems Interconnection - Common Management Information Protocol: Specification", 1997a, International Telecommunications Union ITU-T, Geneva , http://www.itu.int/rec/T-REC-X.711/en [Accessed Sept 2007]

212. (ITU-T 1997b)
ITU-T, "ITU-T recommendation X.901 (1997): Information Technology - Open Distributed Processing Reference Model: Overiview [RM-ODP] (Also published as ISO/IEC 10746 1995)", 1997b, ITU-T, Geneva

213. (ITU-T 1997c)
ITU-T, "ITU-T Recommendation X.901 (1997): Information Technology - Open Distributed Processing Reference Model: Overview [RM-ODP] (Also published as ISO/IEC 10746 1995)", 1997c, ITU-T, Geneva , http://www.itu.int/rec/T-REC-X.901/en

214. (ITU-T 2000a)
ITU-T, "ITU-T G.805 Generic Functional Architecture of Transport Networks Series G: Transmission Systems and Media, Digital Systems and Networks Digital Networks - General Aspects", 2000a, ITU-T, Geneva

215. (ITU-T 2000b)
ITU-T, "ITU-T Recommendation M3000 Series TMN", 2000b, International Telecommunications Union ITU-T, Geneva , http://www.itu.int/rec/T-REC-M.3000/en [Accessed Sept 2007]

216. (ITU-T 2003)
ITU-T, "ITU Recommendation G.709/Y.1331, Interfaces for the Optical Transport Network (OTN), March 2003 (Amendment1 December 2003)", 2003, Geneva

217. (ITU-T 2004)
ITU-T, "ITU-T Recommendation M.3050.0 (07/04) Enhanced Telecommunications Operations Map - Introduction", 2004, http://www.itu.int/rec/T-REC-M.3050.0-200407-S/en

218. (ITU-T 2006)
ITU-T, "ITU-T Study Group 4 NGN Management Focus Group Terms of Reference v4.0 (2nd June 2006)", 2006, International Telecommunications Union ITU-T, Geneva , http://www.itu.int/ITU-T/studygroups/com04/ngn-mfg/docs/tor4_0.doc [Accessed Sept 2007]

219. (ITU-T 2007)
ITU-T, "ITU-T Partnership and External Cooperation Informal Forum Summit", 2007, http://www.itu.int/ITU-T/tsb-director/forum/ [Accessed Sept 2007]

220. (Jaton, Markus 2003a)
Jaton, Markus, "Rapport Scientifique sur le projet ENMP (1 décembre 2003)", 2003a, TCOM Laboratory Swiss Federal Institute of Technology, Lausanne, Switzerland , http://www.iict.ch/Tcom/HomePages/MJN/Projets/ENMP/RapportFinal.pdf [Accessed Sept 2007]

221. (Jaton, Markus 2003b)
Jaton, Markus, "Spécification ENMP/3D-Net Early alpha, status 0.1. 6 Juin 2003", 2003b, TCOM Laboratory Swiss Federal Institute of Technology, Lausanne, Switzerland , http://www.iict.ch/Tcom/HomePages/MJN/Projets/ENMP/Specification.pdf [Accessed Sept 2007]

222. (Jaton, Markus 2002)
Jaton, Markus, "Experimental Network Management Platform Rapport de projet Réserve stratégique HES-SO 11.06.2002", 2002, TCOM Laboratory Swiss Federal Institute of Technology, Lausanne, Switzerland , http://www.iict.ch/Tcom/HomePages/MJN/Projets/ENMP/ENMP.pdf [Accessed Sept 2007]

223. (JBoss 2007c)
JBoss, "JBossMQ Project Homepage", 2007c, http://www.jboss.org/wiki/Wiki.jsp?page=JBossMQ [Accessed Sept 2007]

224. (JBoss 2007a)
JBoss, "JBoss Messaging Project Homepage", 2007a, http://wiki.jboss.org/wiki/Wiki.jsp?page=JBossMessaging [Accessed Sept 2007]

225. (JBoss 2007b)
JBoss, "Drools (Jboss Rules) Project Homepage", 2007b, http://labs.jboss.com/drools/ [Accessed Sept 2007]

226. (JBoss 2007d)
JBoss, "Jboss SNMP Adapter", 2007d, http://wiki.jboss.org/wiki/Wiki.jsp?page=JBossSNMPAdapter [Accessed Sept 2007]

227. (JCP 2001)
JCP, "JSR 053 Java Servlet 2.3 and JavaServer Pages 1.2 Specifications (Final Release 13th August 2001)", 2001, http://jcp.org/aboutJava/communityprocess/final/jsr053/ [Accessed Sept 2007]

228. (JCP 2003a)
JCP, "JSR 12: Java Data Objects (JDO) Specification Final Release 2, 16 Sept. 2003", 2003a, http://www.jcp.org/en/jsr/detail?id=12

229. (JCP 2003b)
JCP, "JSR 168 Java Portlet Specification Final Release 27 Oct, 2003", 2003b, Sun Microsystems, http://jcp.org/en/jsr/detail?id=168 [Accessed Sept 2007]

230. (JCP 2003c)
JCP, "JSR 914: Java Message Service (JMS) API (Final Release 02 Dec, 2003)", 2003c, [Accessed Sept 2007]

231. (JCP 2004a)
JCP, "JCP 2: Process Document The formal procedures for using the Java Specification development process Version 2.6 (March 9, 2004)", 2004a, Sun Microsystems, Inc., http://jcp.org/en/procedures/jcp2 [Accessed Sept 2007]

232. (JCP 2004b)
JCP, "JSR 56: Java Network Launching Protocol and API (Final Release 21 Sep, 2004)", (Java Web Start Architecture), 2004b, Sun Microsystems, http://jcp.org/en/jsr/detail?id=56 [Accessed Sept 2007]

233. (JCP 2004c)
JCP, "JSR 94 Java Rule Engine API Final Release 04 Aug, 2004", 2004c, Java Community Process, http://jcp.org/en/jsr/detail?id=94 [Accessed Sept 2007]

234. (JCP 2006)
JCP, "JSR 927: Java TV API", 2006, Java Community Process, http://jcp.org/en/jsr/detail?id=927

235. (JCP 2007)
JCP, "JSR 142: OSS Inventory API Release 3 Final (27 Aug, 2007)", 2007, http://jcp.org/en/jsr/detail?id=142 [Accessed Sept 2007]

236. (Jetty 2007)
Jetty, "Jetty Java HTTP Servlet Server Project Homepage", 2007, http://jetty.mortbay.org/ [Accessed Sept 2007]

237. (JoeSNMP 2007)
JoeSNMP, "JoeSNMP Project Homepage", 2007, http://sourceforge.net/projects/joesnmp/ [Accessed Sept 2007]

238. (Johnson, Justin Pappas 2002)
Johnson, Justin Pappas, "Open Source Software: Private Provision of a Public Good", Journal of Economics & Management Strategy, 11, [4], 637-662, 2002, http://dx.doi.org/10.1111/j.1430-9134.2002.00637.x

239. (Johnson, R. 2005)
Johnson, R. 2005. *Professional Java development with the Spring Framework*. Indianapolis, IN: Wiley Pub, ISBN.9780764574832 (paper/website)

240. (Josephsen, D. 2007)
Josephsen, D. 2007. *Building a monitoring infrastructure with Nagios*. Upper Saddle River, NJ: Prentice Hall, ISBN.0132236931 (pbk. : alk. paper)

241. (Jrobin 2007)
Jrobin, "Jrobin (Java port of RRDTool) Project Homepage", 2007, http://www.jrobin.org [Accessed Sept 2007]

242. (Junit 2007)
Junit, "Junit Project Homepage", 2007, http://junit.sourceforge.net/ [Accessed Sept 2007]

243. (Kock, N. F. 2007b)
Kock, N. F. 2007b. Preface. In N. F. Kock (Ed.), *Information systems action research: an applied view of emerging concepts and methods*: xix-xxii. New York, N.Y: Springer,.ISBN 9780387360591 (hbk.)

244. (Kock, Ned F. 2004)
Kock, Ned F., "The three threats of action research: a discussion of methodological antidotes in the context of an information systems study", Decis.Support Syst., 37, [2], 265-286, 2004, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands , http://dx.doi.org/10.1016/S0167-9236(03)00022-8

245. (Kock, N. F. 2007a)
Kock, N. F. 2007a. *Information Systems Action Research: an applied view of emerging concepts and methods*. New York, N.Y: Springer, ISBN.9780387360591 (hbk.) , http://dx.doi.org/10.1007/978-0-387-36060-7

246. (Kock, Ned F., McQueen, R. J., Baker, M., and Rouse, A. 1996)

Kock, Ned F., McQueen, R. J., Baker, M., and Rouse, A., "Negotiation in information systems action research", ISCNZ '96: Proceedings of the 1996 Information Systems Conference of New Zealand (ISCNZ '96), 164, 1996, IEEE Computer Society, Washington, DC, USA 0-8186-7710-4, http://doi.ieeecomputersociety.org/10.1109/ISCNZ.1996.555292

247. (Koziol, J. 2003)
Koziol, J. 2003. *Intrusion detection with Snort*. Indianapolis, Ind: Sams, ISBN.157870281X

248. (Kretchmar, J. M. 2004)
Kretchmar, J. M. 2004. *Open source network administration*. Upper Saddle River, N.J: Prentice Hall Professional Technical Reference, ISBN.0130462101

249. (Kuan, J 2002)
Kuan, J, "Open source software as lead-user's make or buy decision: A study of open and closed source quality [Working paper]", 2002, Stanford Institute of Economic Policy Research, Stanford University, Stanford, CA. http://idei.fr/doc/conf/sic/papers_2002/osskuan.pdf [Accessed Sept 2007]

250. Kuk, G. 2006. Strategic Interaction and Knowledge Sharing in the KDE Developer Mailing List. *Management Science*, 52(7): 1031-1042.

251. (Kwan, Stephen K. and West, Joel 2005)
Kwan, Stephen K. and West, Joel, "A Conceptual Model for Enterprise Adoption of Open Source Software.", Bolin, Sherrie (Eds.), The Standards Edge: Open Season, 274-301, 2005, Sheridan Books, Ann Arbor, Mich. http://www.cob.sjsu.edu/OpenSource/Research/KwanWest2005.pdf [Accessed Sept 2007]

252. (Ladd, Seth, Davison, Darren, Devijver, Steven, and Yates, Colin 2006)
Ladd, Seth, Davison, Darren, Devijver, Steven, and Yates, Colin, "Expert Spring MVC and Web Flow", Harrop, Rob and Donald, Keith (Eds.), 2006, AP Press, Berkley, CA ISBN: 1-59059-584-X

253. (Lakani, K. R. & Wolf, R. G. 2005)
Lakani, K. R. & Wolf, R. G. 2005. Why Hackers Do What They Do - Understanding Motivation and Effort in Free/Open Source Software Projects. In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*: 3-21. Cambridge, Mass: MIT Press,.ISBN 0-262-06246-1

254. (Lapierre, Brigitte Hénault 1996)
Lapierre, Brigitte Hénault, "Bidirectional Information Transfer: An Imperative for Network and Marketing Integration in a Canadian Telecommunications Firm", Journal of Product Innovation Management, 13, [2], 152-166, 1996, Elsevier 0737-6782, http://dx.doi.org/10.1111/1540-5885.1320152

255. (Latimore, Dan 2002)
Latimore, Dan, "Calculating value during uncertainty: Getting real with "real options"", 2002, IBM Institute for Business Value, http://www-935.ibm.com/services/uk/igs/pdf/esr-calculating-value-during-uncertainty.pdf [Accessed Sept 2007]

256. (Lau, F. 1997)
Lau, F., "A review on the use of action research in information systems studies", Proceedings of the IFIP TC8 WG 8.2 international conference on Information systems and qualitative research, 31-68, 1997, Chapman & Hall, Ltd., London, UK, UK 0-412-82360-8, http://portal.acm.org/citation.cfm?id=278888.278894&coll=GUIDE&dl=ACM&CFID=15151515&CFTOKEN=6184618# Philadelphia, Pennsylvania, United States,

257. (Lau, F. 1999)
Lau, F., "Toward a framework for action research in information systems studies", Information Technology & People, 12, [1], 148-176, 1999, Emerald Group Publishing Limited, London, UK 0959-3845, http://dx.doi.org/10.1108/09593849910267206

258. (Laurent, A. M. 2004)
Laurent, A. M. 2004. *Understanding open source and free software licensing*. Beijing: O'Reilly, ISBN.0596005814 (pbk.)

259. (Lawton, G. 2002)
Lawton, G., "Open Source Security: Opportunity or Oxymoron?", IEEE Computer, 35, [3], 18-21, 2002, http://dx.doi.org/10.1109/2.989921

260. (Le Claire, Jennifer 2007)
Le Claire, Jennifer, "Nokia Readies Challenge to iTunes (Thu Aug 30 2007)", newsfactor.com, 2007, Yahoo News, http://news.yahoo.com/s/nf/20070830/bs_nf/55007 [Accessed Sept 2007]

261. (Lee, J. & Ware, B. 2003)
Lee, J. & Ware, B. 2003. *Open source Web development with LAMP using Linux, Apache, MySQL, Perl, and PHP*. Boston: Addison-Wesley, ISBN.020177061X (pbk. : alk. paper)

262. (Lee, John and Ben Natan, Ron 2002)
Lee, John and Ben Natan, Ron, "Integrating service level agreements", : optimizing your OSS for SLA delivery, 2002, Indianapolis, Ind. ; [Great Britain] : Wiley Pub 0471210129 (pbk) ORDERED

263. (Leenders, Mark A. A. M. and Wierenga, Bernard 2002)
Leenders, Mark A. A. M. and Wierenga, Bernard, "The effectiveness of different mechanisms for integrating marketing and R&D", Journal of Product Innovation Management, 19, [4], 305-317, 2002, Elsevier 0737-6782, http://dx.doi.org/10.1111/1540-5885.1940305

264. (Lerner, J. & Tirole, J. 2005)

Lerner, J. & Tirole, J. 2005. Economic Perspectives on Open Source. In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*: 47-78. Cambridge, Mass: MIT Press,.ISBN 0-262-06246-1

265. (Levitt, Theodore 1960)
Levitt, Theodore, "Marketing myopia", Harvard Business Review, 38, [4], 45-56, 1960, Harvard Business School, http://www.carreirasolo.org/archives/arquivos/MarketingMyopia.pdf [Accessed Sept 2007]

266. (Lewis, David, Wade, Vincent, and Cullin, Brian 2003)
Lewis, David, Wade, Vincent, and Cullin, Brian, "Towards the Technology Neutral Modelling of Management Components", Journal of Network and Systems Management, 11, [1], 39-56, 2003, Springer, Netherlands , http://dx.doi.org/10.1023/A:1022493024547

267. (Li, Jingyue, Bjørnson, Finn Olav, Conradi, Reidar, and Kampenes, Vigdis By 2004)
Li, Jingyue, Bjørnson, Finn Olav, Conradi, Reidar, and Kampenes, Vigdis By, "An Empirical Study of Variations in COTS-based Software Development Processes in Norwegian IT Industry", Proceedings of the 10th IEEE International Metrics Symposium, Chicago, USA, September, 2004, 72-83, 2004, IEEE CS Press, http://dx.doi.org/10.1109/METRIC.2004.1357892

268. (Li, Jingyue, Reidar, Conradi, Slyngstad, Odd Petter N., Bunse, Christian, Khan, Umair, Torchiano, Marco, and Morisio, Maurizio 2007)
Li, Jingyue, Reidar, Conradi, Slyngstad, Odd Petter N., Bunse, Christian, Khan, Umair, Torchiano, Marco, and Morisio, Maurizio, "An Empirical Study on Off-the-Shelf Component Usage in Industrial Projects", Proceedings of the 6th International Conference on Product Focused Software Process Improvement (PROFES'2005), June 13-16 2005, Oulu, Finland., LNCS, 3547/2005, 54-68, 2007, Springer, Berlin / Heidelberg ISBN 978-3-540-26200-8, http://dx.doi.org/10.1007/11497455_7

269. (Li, Sing 2002)
Li, Sing, "From black boxes to enterprises, Part 3: Hands-on JMX integration Connecting JMX Agent to a real-life network management system", IBM Developer Works, 2002, IBM, http://www.ibm.com/developerworks/java/library/j-jmx3/ [Accessed Sept 2007]

270. (LibertyAlliance 2007)
LibertyAlliance, "Liberty Alliance Homepage", 2007, http://www.projectliberty.org/ [Accessed Sept 2007]

271. (Locke, J. 2004)
Locke, J. 2004. *Open source solutions for small business problems*. Hingham, Mass: Charles River Media, ISBN.1584503203 (pbk. with cd-rom : alk. paper)

272. (Magedanz, T., Witaszek, D., and Knuettel, K. 2005)
Magedanz, T., Witaszek, D., and Knuettel, K., "The IMS playground @ FOKUS-an open testbed for generation network multimedia services", First International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities 2005 (Tridentcom 2005), 2-11, 2005, IEEE, http://dx.doi.org/10.1109/TRIDNT.2005.35

273. (Mannaert, Herwig and Adriaenssens, Paul 2006)
Mannaert, Herwig and Adriaenssens, Paul, "A Web-Based Operational Support System for Digital Movie Distribution", International Conference on Internet and Web Applications and Services/Advanced International Conference on Telecommunications, 2006.(AICT-ICIW '06), 192, 2006, 0-7695-2522-9, http://dx.doi.org/10.1109/AICT-ICIW.2006.40

274. (Mannaert, Herwig and Adriaenssens, Paul 2004)
Mannaert, Herwig and Adriaenssens, Paul, "Exploring Service Reliability for Content Distribution to Partial or Intermittent DVB-S Satellite Receivers", Service Assurance with Partial and Intermittent Resources, LNCS, 104-109, 2004, Springer, Berlin ISBN 978-3-540-22567-6, http://dx.doi.org/10.1007/b99076

275. (Mannaert, Herwig and Adriaenssens, Paul 2005a)
Mannaert, Herwig and Adriaenssens, Paul, "A scheduling engine to allocate resources for multicast transmissions over a satellite link", Proceedings of Advanced Industrial Conference on Telecommunications/Service Assurance with Partial and Intermittent Resources Conference / E-Learning on Telecommunications Workshop.AICT/SAPIR/ELETE 2005, 377-381, 2005a, http://dx.doi.org/10.1109/AICT.2005.11

276. (Mannaert, Herwig and Adriaenssens, Paul 2005b)
Mannaert, Herwig and Adriaenssens, Paul, "Web services based systems for network management and provisioning: a case study", Proceedings of the Advanced Industrial Conference on Telecommunications/Service Assurance with Partial and Intermittent Resources Conference / E-Learning on Telecommunications Workshop 17-20 July 2005 (AICT/SAPIR/ELETE 2005), 442-445, 2005b, http://dx.doi.org/10.1109/AICT.2005.94

277. (Mannaert, Herwig, Adriaenssens, Paul, and Meyers, E. 2006)
Mannaert, Herwig, Adriaenssens, Paul, and Meyers, E., "ARTES 4 Project Final Report: Asset and Network Management for D-Cinema", 2006, European Space Agency, http://telecom.esa.int/telecom/www/object/index.cfm?fobjectid=23083 [Accessed Sept 2007]

278. (Mannaert, Herwig and Ven, Kris 2005)
Mannaert, Herwig and Ven, Kris, "The use of open source software platforms by Independent Software Vendors: issues and opportunities", Proceedings of the fifth workshop on Open source software engineering (WOSSE), 1-4, 2005, ACM Press, New York, NY, USA 1-59593-127-9, http://doi.acm.org/10.1145/1083258.1083266 St. Louis, Missouri

279. (Markowitz, H. M. 1959)
Markowitz, H. M. 1959. *Portfolio selection - efficient diversification of investments*. New York: Wiley

280. (Marson, Ingrid 2004)
Marson, Ingrid, "Sybase releases free database for Linux", 2004, CNET News.com, http://news.com.com/Sybase+releases+free+database+for+Linux/2100-7344_3-5359441.html [Accessed Sept 2007]

281. (Marssol, Vincent and van Zyl, Jason 2006)
Marssol, Vincent and van Zyl, Jason, "Better Builds with Maven - How-to Guide for Maven 2.0 (Book Version 1.0.1)", 2006, Mergere Library Press, USA , http://www.devzuz.com/web/guest/products/resources#BBWM [Free Download - Accessed Sept 2007]

282. (Matusow, J. 2005)
Matusow, J. 2005. Shared Source: The Microsoft Perspective. In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*: 330-346. Cambridge, Mass: MIT Press,.ISBN 0-262-06246-1, http://mitpress.mit.edu/catalog/item/default.asp?ttype=2&tid=10477&mode=toc

283. (McArdle, K. L. & Reason, P. 2007)
McArdle, K. L. & Reason, P. 2007. Action Research and Organization Development. In T. Cummings (Ed.), *Handbook of Organization Development (forthcoming 2007)*: Thousand Oaks, CA: Sage Publications,.ISBN 9780761928126, http://people.bath.ac.uk/mnspwr/Papers/ActionResearch&OrganizationDevelopment.pdf [Accessed Sept 2007]

284. (McKay, J. & Marshal, P. 2007)
McKay, J. & Marshal, P. 2007. Driven by Two Masters, Serving Both - The Interplay of Problem Solving and Research in Information Systems Action Research Projects. In N. F. Kock (Ed.), *Information systems action research: an applied view of emerging concepts and methods*: 151-158. New York, N.Y: Springer,.ISBN 9780387360591 (hbk.), http://dx.doi.org/10.1007/978-0-387-36060-7_6

285. (McKay, Judy and Marshal, Peter 2002)
McKay, Judy and Marshal, Peter, "Action Research: a Guide to Process and Procedure.", European Conference on Research Methods in Business & Management, University of Reading, April 29-30 2002, 2002, University of Tasmania, Australia , http://www.utas.edu.au/infosys/publications/research/Qual_Research/mckay%20&%20marshall_%20Action%20Research%20process%20model.pdf [Accessed Sept 2007]

286. (McKay, Judy and Marshal, Peter 2000)
McKay, Judy and Marshal, Peter, "Quality and Rigour of Action Research in Information Systems.;", Proceedings of the Eighth European Conference on Information Systems, Vienna (Hansen HR, Bichler M, Mahrer H eds.), 108-115, 2000, http://csrc.lse.ac.uk/asp/aspecis/20000011.pdf [Accessed Sept 2007]

287. (Microsoft 2007)
Microsoft, "Windows Management Instrumentation (WMI)", 2007, http://msdn2.microsoft.com/en-us/library/aa384642.aspx [Accessed Sept 2007]

288. (Milham, D and Ronco, E 2004)
Milham, D and Ronco, E, "How can the eTOM framework help service providers in today's marketplace?", Network Operations and Management Symposium, 2004.NOMS 2004.IEEE/IFIP, 1, 59-71, 2004, IEEE http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1317837

289. (Milham, D. 2005)
Milham, D., "Fixed Mobile Service Convergence: If we make it, can we manage it?", Strategic OSS for Next Generation Operations, London 17-18 January 2005, 2005, OQPC, London

290. (Miller, R. B., Heiman, S. E., & Tuleja, T. 2005)
Miller, R. B., Heiman, S. E., & Tuleja, T. 2005. *The new strategic selling - the unique sales system proven successful by the world's best companies*. New York, NY: Warner Books, ISBN.044669519X (pbk.)

291. (Mills, J. A. and Ruston, L. 1990)
Mills, J. A. and Ruston, L., "The OSCA architecture: enabling independent product software maintenance", Proceedings of Euromicro '90 Workshop on Real Time, 6-8 June 1990, 64-70, 1990, IEEE, http://dx.doi.org/10.1109/EMWRT.1990.128229

292. (Mintzberg, H., Ahlstrand, B. W., & Lampel, J. 2001)
Mintzberg, H., Ahlstrand, B. W., & Lampel, J. 2001. *Strategy safari the complete guide through the wilds of strategic management*. Harlow: Financial Times Prentice Hall, ISBN.0273656368

293. (MIS Quarterly 2004)
MIS Quarterly 2004. *MIS Quarterly Special Issue on Action Research in Information Systems, September 2004*. Minneapolis: University of Minnesota, http://www.misq.org/archivist/vol/no28/Issue3/vol28n3index.html

294. (Mobicents 2007)
Mobicents, "Mobicents Open SLEE", 2007, http://wiki.java.net/bin/view/Communications/MobicentsOpenSLEE [Accessed Sept 2007]

295. (Mockus, A., Fielding, R. T., & Herbsleb, J. D. 2005)

Mockus, A., Fielding, R. T., & Herbsleb, J. D. 2005. Two Case Studies of Open Source Software Development: Apache and Mozilla. In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*: 164-209. Cambridge, Mass: MIT Press,.ISBN 0-262-06246-1

296. (Moody, Glyn 2006)
Moody, Glyn, "Google: the Godfather of Open Source?", 2006, http://www.linuxjournal.com/node/1000076 [Accessed Sept 2007]

297. (Moore, G. A. 1999)
Moore, G. A. 1999. *Crossing the chasm marketing and selling high-tech products to mainstream customers*. New York: HarperBusiness, ISBN.0066620023

298. (Morisio, Maurizio and Torchiano, Marco 2002)
Morisio, Maurizio and Torchiano, Marco, "Definition and Classification of COTS: A Proposal", Proceedings of First International Conference on COTS-Based Software Systems (ICCBSS 2002), Orlando, FL, USA, February 4-6, 2002., LNCS, 2255/2002, 165-175, 2002, Springer, Berlin / Heidelberg ISSN 0302-9743, http://www.springerlink.com/content/ej3qdydn3h2p53bb/?p=9c6285dabaec4981adc64860aeba5534&pi=0

299. (Mountjoy, Jon 2006)
Mountjoy, Jon, "Dev2Dev Interview Series: Kodo - Towards an Open Source EJB 3.0 Persistence Engine
Neelan Choksi interviewed by Jon Mountjoy 14th Jan 2006", 2006, BEA, http://dev2dev.bea.com/pub/a/2006/02/interview-kodo-opensource.html [Accessed Sept 2007]

300. (Mozilla 2007)
Mozilla, "Mozilla Firefox Web Browser", 2007, Mozilla Project, http://www.mozilla.com/firefox/ [Accessed Sept 2007]

301. (Mumford, Enid 2001)
Mumford, Enid, "Advice for an action researcher", Journal: Information Technology & People, 14, [1], 12-27, 2001, MCB UP Ltd, UK 0959-3845, http://dx.doi.org/10.1108/09593840110384753

302. (Murch, R. 2004)
Murch, R. 2004. Autonomic Computing and Open Standards. *Autonomic Computing*: Upper Saddle River, N.J: Prentice Hall Professional Technical Reference,.ISBN 013144025X (alk. paper)

303. (Mustonen, Mikko 2005)
Mustonen, Mikko, "When Does a Firm Support Substitute Open Source Programming?", Journal of Economics & Management Strategy, 14, [1], 121-139, 2005, http://www.blackwell-synergy.com/doi/abs/10.1111/j.1430-9134.2005.00036.x

304. (MX4J 2007)
MX4J, "MX4J Open Source JMX Library Project Homepage", 2007, http://mx4j.sourceforge.net/ [Accessed Sept 2007]

305. (Nagios 2007)
Nagios, "Nagios project Homepage", 2007, http://www.nagios.org/ [Accessed Sept 2007]

306. (NetBeans 2007)
NetBeans, "NetBeans IDE Homepage", 2007, http://www.netbeans.org/ [Accessed Sept 2007]

307. (NGW 2007)
NGW, "National Grid Wireless Plc", 2007, http://www.nationalgridwireless.com/ [Accessed Sept 2007]

308. (Nistnet 2007)
Nistnet, "Nistnet", 2007, http://www-x.antd.nist.gov/nistnet/ [Accessed Sept 2007]

309. (NMRG 2003)
NMRG, "SMIng Next Generation Structure of Management Information", 2003, http://www.ibr.cs.tu-bs.de/projects/sming/ [Accessed Sept 2007]

310. (NMRG 2006a)
NMRG, "Minutes of the joint EMANICS / IRTF-NMRG Workshop on Future Direction of Network and Service Management Research 19-20 October 2006, Utrecht, The Netherlands", 2006a, http://www.ibr.cs.tu-bs.de/projects/nmrg/minutes/minutes-021.txt [Accessed Sept 2007]

311. (NMRG 2006b)
NMRG, "Participants Prepared Statements for the EMANICS / IRTF-NMRG Workshop on Future Direction of Network and Service Management Research 19-20 October 2006, Utrecht, The Netherlands", 2006b, http://www.ibr.cs.tu-bs.de/projects/nmrg/meetings/2006/utrecht/nmrg-21-reader.pdf [Accessed Sept 2007]

312. (Nokia 2007)
Nokia, "Open EMS Suite by Nokia: a multi-technology EMS platform", 2007, http://www.nokia.com/link?cid=PLAIN_TEXT_6406 [Accessed Sept 2007]

313. (Norris, J. S. and Kamp, P. H. 2004)
Norris, J. S. and Kamp, P. H., "Mission-critical development with open source software: Lessons learned", IEEE Software, 21, [1], 42-49, 2004, http:/dx.doi.org/10.1109/MS.2004.1259211

314. (NwSysMgt 2007)
NwSysMgt, "Journal of Network and Systems Management", 2007, Springer, Netherlands 1573-7705, http://www.springerlink.com/content/1573-7705/ [Accessed Sept 2007]

315. (O'Donnell, Shane 2001a)
O'Donnell, Shane, "OpenNMS Update Vol 2., Issue 1 Jan. 2, 2001", 2001a, OpenNMS, http://lwn.net/2001/0104/a/opennms-0103.php3 [Accessed Sept 2007]

316. (O'Donnell, Shane 2001b)
O'Donnell, Shane, "OpenNMS Update Vol 2., Issue 2 Jan. 9, 2001", 2001b, OpenNMS, http://old.lwn.net/2001/0111/a/opennms-0109.php3 [Accessed Sept 2007]

317. (O'Donnell, Shane 2000b)
O'Donnell, Shane, "Open Source Tools in Network Management: Waking a Sleeping Giant", 2000b, Linux Users Group at North Carolina State University, http://lug.ncsu.edu/lectures/opennms/ [Accessed Sept 2007]

318. (O'Donnell, Shane 2000a)
O'Donnell, Shane, "Network management: open source solutions to proprietary problems", Proceedings of the 28th annual ACM SIGUCCS conference on User services: Building the future, 208-217, 2000a, ACM Press, New York, NY, USA ISBN:1-58113-299-8, http://doi.acm.org/10.1145/354908.354956, (Available at http://www.shaneo.com/prof/OpenSourceNetworkManagementTools.pdf) [Accessed Sept 2007]

319. (O'Donnell, Shane 2002)
O'Donnell, Shane, "Network Management With OpenNMS", ONLamp.com, 2002, O'Reilly, http://www.onlamp.com/pub/a/onlamp/2002/04/18/opennms.html?page=1 [Accessed Sept 2007]

320. (Ochs, M. A., Pfahl, D., and Chrobok-Diening, G. 2001)
Ochs, M. A., Pfahl, D., and Chrobok-Diening, G., "A Method for Efficient Measurement-based COTS Assessment and Selection: Method Description and Evaluation Results", Proceedings of IEEE 7th International Software Metrics Symposium, 4-6 April 2001, London, UK, 285-296, 2001, http://dx.doi.org/10.1109/METRIC.2001.915536

321. (OFCOM 20-12-2006)
OFCOM, "The Communications Market: Digital Progress Report Digital TV, Q3 2006", 20-12-2006, OFCOM, London , http://www.ofcom.org.uk/research/tv/reports/dtv/dtu_2006_q3/dtu_2006_q3.pdf [Accessed Sept 2007]

322. (OGC 2002)
OGC, "Office of Government Commerce, Introduction to PRINCE2 (PRojects IN Controlled Environments)", 2002, Office of Government Commerce, UK , http://www.ogc.gov.uk/methods_prince_2__overview.asp [Accessed Sept 2007]

323. (OMG 2002)
OMG, "Meta-Object Facility (MOF™), version 1.4", 2002, http://www.omg.org/technology/documents/formal/mof.htm [Accesssed Sept 2007]

324. (OMG 2007a)
OMG, "OMG Model Driven Architecture (MDA) Guide", 2007a, http://www.omg.org/mda/specs.htm#MDAGuide [Accessed Sept 2007]

325. (OMG 2007b)
OMG, "OMG's CORBA Website Homepage", 2007b, Object Management Group, http://www.corba.org/ [Accessed Sept 2007]

326. (Open-MHP 2007)
Open-MHP, "Open-MHP (Open Source DVB-MHP Digital Video Broadcasting – Multimedia Home Platform)", 2007, http://www.openmhp.org/ [Accessed Sept 2007]

327. (OpenDRIM 2007)
OpenDRIM, "OpenDRIM Project Overview", 2007, http://www.opendrim.org/opendrim/attachment/wiki/1stOpenDRIMDeveloperCamp/1-OpenDrimProposal2-gcng-061030.pdf [Accessed Sept 2007]

328. (OpenESB 2007)
OpenESB, "Open ESB Project Homepage", 2007, https://open-esb.dev.java.net/ See Also Sun description http://java.sun.com/integration/index.jsp [Accessed Sept 2007]

329. (OpenNMS 2007a)
OpenNMS, "How-To Get Started with OpenNMS (Parts 2-5)", 2007a, http://sourceforge.net/docman/index.php?group_id=4141 [Accessed Sept 2007]

330. (OpenNMS 2007b)
OpenNMS, "OpenNMS Installation Guide", 2007b, http://www.opennms.org/documentation/installguide.html [Accessed Sept 2007]

331. (OpenNMS 2007d)
OpenNMS, "OpenNMS Release Notes", 2007d, http://www.opennms.org/documentation/ReleaseNotesStable.html [Accessed Sept 2007]

332. (OpenNMS 2007c)
OpenNMS, "OpenNMS Project Homepage", 2007c, www.opennms.org [Accessed Sept 2007]

333. (OpenOffice 2007)
OpenOffice, "OpenOffice Project Homepage", 2007, http://www.openoffice.org/ [Accessed Sept 2007]

334. (OpenOSS 2007)
OpenOSS, "Open Source Operational Support System", 2007, http://sourceforge.net/projects/openoss [Accessed Sept 2007]

335. (OpenPKGOpenPKG, "OpenPKG Open Package Project History", http://www.openpkg.org/project/history.php [Accessed Sept 2007]

336. (OpenWBEM 2007)
OpenWBEM, "OpenWBEM", 2007, http://www.openwbem.org/

337. (OSGi 2007)
OSGi, "The OSGi Alliance Homepage", 2007, http://www.osgi.org/ [Accessed Sept 2007]

338. (OSI 2006)
OSI, "The Open Source Definition (Annotated)", 2006, Open Source Initiative, http://www.opensource.org/docs/definition.php [Accessed Sept 2007]

339. (OSS/J 2002a)
OSS/J, "JSR 091: OSS/J OSS Trouble Ticket API (Final Release 19 Feb, 2002)", 2002a, http://jcp.org/en/jsr/detail?id=91 [Accessed Sept 2007]

340. (OSS/J 2002b)
OSS/J, "JSR 90: OSS Quality of Service API", 2002b, http://jcp.org/en/jsr/detail?id=90

341. (OSS/J 2002c)
OSS/J, "JSR 90: OSS Quality of Service API (18 Nov, 2002)", 2002c, http://jcp.org/en/jsr/detail?id=90 [Accessed Sept 2007]

342. (OSS/J 2006a)
OSS/J, "Core Business Entities Concepts and Principles", TMFC4778 CBE, 2006a, http://www.tmforum.org/browse.aspx?catid=2951&linkID=6390&docID=4778 [Accessed Sept 2007]

343. (OSS/J 2006b)
OSS/J, "OSS through Java Web Services Integration Profile (WSIP) Version 1.0", OSS through Java™ Initiative, 2006b, www.ossj.org

344. (OSS/J 2007a)
OSS/J, "JSR 144: OSS/J OSS Common API (Maintenance Release 5 24 Jul, 2007)", (Core Business Entities, CBE API), 2007a, http://jcp.org/en/jsr/detail?id=144 [Accessed Sept 2007]

345. (OSS/J 2007b)
OSS/J, "OSS Through Java (OSS/J)", 2007b, TMForum, www.ossj.org [Accessed Sept 2007]

346. (OSS/J 2007c)
OSS/J, "Service and Business Management - OSS/J Open Source on java.net", 2007c, https://sbm.dev.java.net/ [Accessed Sept 2007]

347. (OSSbeans 2007)
OSSbeans, "OSSbeans Project Homepage", 2007, University of Southampton, http://openoss.sourceforge.net [Accessed Sept 2007]

348. (OSSIM 2007)
OSSIM, "Open Source Security Information Management (OSSIM) Project Homepage", 2007, http://www.ossim.net/docs.php [Accessed Sept 2007]

349. (OTRS 2007)
OTRS, "Open Ticket Request System (otrs)", 2007, www.otrs.org [Accessed Sept 2007]

350. (OW2 2007a)
OW2, "Objectweb", 2007a,

351. (OW2 2007b)
OW2, "ObjectWeb 2 Consortium Homepage", 2007b, http://aboutus.ow2.org/ [Accessed Sept 2007]

352. (Page, V., Dixon, M., and Choudhury, I. 2007)
Page, V., Dixon, M., and Choudhury, I., "Security risk mitigation for information systems", BT Technology Journal, 25, 118-127, 2007, Kluwer Academic Publishers, Hingham, MA, USA , http://dx.doi.org/10.1007/s10550-007-0014-8

353. (Pavlou, George, Fegklas, Paris, Goorveris, Stelios, and Liotta, Antonio 2004)
Pavlou, George, Fegklas, Paris, Goorveris, Stelios, and Liotta, Antonio, "On management technologies and the potential of Web services", IEEE Communications Magazine, 42, [7], 58-66, 2004, IEEE, http://dx.doi.org/10.1109/MCOM.2004.1316533

354. (Pereira, José Pedro, de Matos, Valente, Chainho, Paulo, Lin, Guochun, and Mäkinen, Susanna 2001)
Pereira, José Pedro, de Matos, Valente, Chainho, Paulo, Lin, Guochun, and Mäkinen, Susanna, "Eurescom Study P1044 Report (T1): Aspects of Open Source in Telecommunications - Technical Evaluation of Open Source Products", Aspects of Open Source in Telecommunications, 2001, Eurescom, Heidelberg, Germany , http://www.eurescom.de/Public/projects/p1000-series/P1044/default.asp [Accessed Sept 2007]

355. (Pfleging, B. & Zetlin, M. 2006)
Pfleging, B. & Zetlin, M. 2006. *The geek gap - why business and technology professionals don't understand each other and why they need each other to survive.* Amherst, N.Y: Prometheus Books, ISBN.1591024153 (hardcover : alk. paper) , http://www.loc.gov/catdir/toc/ecip0610/2006008263.html

356. (Piper, Andy, Hsiao, Eric, Johndon, Rod, and Wall, Chris 2007)
Piper, Andy, Hsiao, Eric, Johndon, Rod, and Wall, Chris, "Spring 2.0.1 and BEA WebLogic Server 9.2 Integration", Dev2Dev, 2007, BEA, http://dev2dev.bea.com/pub/a/2007/04/spring-2-weblogic-server-9-integration.html [Accessed Sept 2007]

357. (Polanyi, M. 1967)

Polanyi, M. 1967. *The tacit dimension*. London: Routledge & K. Paul

358. (Porter, M. E. 1980)
Porter, M. E. 1980. *Competitive strategy - techniques for analyzing industries and competitors*. New York: Collier Macmillan, ISBN.0029253608

359. (Porter, M. E. 1998)
Porter, M. E. 1998. *Competitive advantage - creating and sustaining superior performance : with a new introduction*. New York: Free Press, ISBN.0684841460

360. (Porter, M. E. 1990)
Porter, M. E. 1990. *The competitive advantage of nations*. New York: Free Press, ISBN.0029253616

361. (PostgreSQL 2007)
PostgreSQL, "PostgreSQL Database Project Homepage", 2007, http://www.postgresql.org/ [Accessed Sept 2007]

362. (Potter, J. M. M. and Brady, A. R. 2005)
Potter, J. M. M. and Brady, A. R., "Strategic changes and frameworks affecting future OSS/BSS in the telecommunications industry", BT Technology Journal, 23, [3], 30-44, 2005, Springer, Netherlands ISSN 1573-1995, http://dx.doi.org/10.1007/s10550-005-0028-z

363. (Pras, A and Schoenwaelder, J 2003)
Pras, A and Schoenwaelder, J, "RFC3444: On the Difference between Information Models and Data Models", 2003, IETF, http://www.ietf.org/rfc/rfc3444.txt [Accessed Sept 2007]

364. (Pras, Aiko, Schonwalder, Jurgan, and Festor, Oliver 2004)
Pras, Aiko, Schonwalder, Jurgan, and Festor, Oliver, "Guest editorial - XML-based management of networks and services", IEEE Communications Magazine, 42, [7], 56-57, 2004, IEEE, http://dx.doi.org/10.1109/MCOM.2004.1316532

365. (Proctor, Mark, Neale, Michael, Frandsen, Michael, Griffith Jr., Sam, Tirelli Edson, and Meyer, Fernando 2007)
Proctor, Mark, Neale, Michael, Frandsen, Michael, Griffith Jr., Sam, Tirelli Edson, and Meyer, Fernando, "JBoss Rules User Guide 4.0.0 GA 24-Jul-2007", 2007, Jboss, http://labs.jboss.com/drools/documentation.html [Accessed Sept 2007]

366. (Proper, H. A., Verrijn-Stuart, A. A., and Hoppenbrouwers, S. J. B. A. 2005)
Proper, H. A., Verrijn-Stuart, A. A., and Hoppenbrouwers, S. J. B. A., "On utility-based selection of architecture-modelling concepts", APCCM '05: Proceedings of the 2nd Asia-Pacific conference on Conceptual modelling, 25-34, 2005, Australian Computer Society, Inc, Darlinghurst, Australia, Australia , http://portal.acm.org/citation.cfm?id=1082276.1082280

367. (Puschmann, T. and Alt, R. 2004)
Puschmann, T. and Alt, R., "Process portals - architecture and integration", Proceedings of the 37th Annual Hawaii International Conference on Systems Sciences 2004, 2004, IEEE, NJ , http://dx.doi.org/10.1109/HICSS.2004.1265552

368. Rapoport, R. N. 1970. Three Dilemmas in Action Research: With Special Reference to the Tavistock Experience. *Human Relations*, 23(6): 499-513.

369. (Rasmussen, Baungaard and Nielsen, Thorkild 2004)
Rasmussen, Baungaard and Nielsen, Thorkild, "Entrepreneurial capabilities: Is entrepreneurship action research in disguise?", AI & Society, 18, [2], 100-112, 2004, Springer-Verlag, London, UK 0951-5666, http://dx.doi.org/10.1007/s00146-003-0270-1

370. (Raymond, Eric S. 1999)
Raymond, Eric S., "The Cathedral and the Bazaar", : musings on Linux and open source by an accidental revolutionary, 1999, Beijing ; Cambridge : O'Reilly, 1999 1565927249, http://www.catb.org/~esr/writings/cathedral-bazaar/

371. (Reddy, N. Mohan, Aram, John D., and Lynn, Leonard H. 1991)
Reddy, N. Mohan, Aram, John D., and Lynn, Leonard H., "The Institutional Domain of Technology Diffusion", Journal of Product Innovation Management, 8, [4], 295-304, 1991, Elsevier 1540-5885, http://dx.doi.org/10.1111/1540-5885.840295

372. (Reeve, M. H., Bilton, C, Homes, P. E., and Bross, M. 2005)
Reeve, M. H., Bilton, C, Homes, P. E., and Bross, M., "Networks and systems for BT in the 21st century", BT Technology Journal, 2005, Springer, Netherlands ISSN 1573-1995, http://dx.doi.org/10.1007/s10550-005-0103-5

373. (Reilly, John P. and Creaner, Martin J. 2005)
Reilly, John P. and Creaner, Martin J., "NGOSS distilled", 2005, The Lean Corporation (Telemanagement Forum) ISBN 0-9551087-0-5

374. (Reilly, Robert 2007)
Reilly, Robert, "Splunk at FreshDirect", 2007, Splunk Inc., http://www.splunk.com/solutions/2017 [Accessed Sept 2007]

375. (Reynolds, Peter, Thorogood, Alan, and Yetton, Phillip 2005)
Reynolds, Peter, Thorogood, Alan, and Yetton, Phillip, "A $1 billion Action Research IT project: building a "new" bank", WISICT '05: Proceedings of the 4th international symposium on Information and communication technologies, 38-43, 2005, Trinity College Dublin, Dublin 1-59593-169-4, http://portal.acm.org/citation.cfm?id=1071761&coll=Portal&dl=GUIDE&CFID=13172757&CFTOKEN=32602789#

376. (Riege, Andreas M. 2003)
Riege, Andreas M., "Validity and reliability tests in case study research: a literature review with "hands-on" applications for each research phase", Qualitative Market Research: An International Journal, 6, [2], 75-86, 2003, MCB UP Ltd, UK 1352-2752, http://dx.doi.org/10.1108/13522750310470055

377. (ROCKET 2003)
ROCKET, "State-of-the-art of Knowledge Management, Deliverable D2.1 from Roadmap to Communicating Knowledge Essential for the indusTrial environment (ROCKET), IST-2001-38245", 2003, European Union IST program, http://rocket.vub.ac.be/public_drafts/ROCKET-D2.1-final-v1.pdf [Accessed Sept 2007]

378. (Rogers, E. M. 2003)
Rogers, E. M. 2003. *Diffusion of innovations*. New York: Free Press, ISBN.0743222091 , http://www.loc.gov/catdir/bios/simon052/2003049022.html;http://www.loc.gov/catdir/enhancements/fy0631/2003049022-t.html;http://www.loc.gov/catdir/enhancements/fy0641/2003049022-s.html

379. (Rose, L. C. 2003)
Rose, L. C., "Risk Management of COTS based System development", Cechich, A., Piattini, M., and Vallecillo, A. (Eds.), Component-Based Software Quality - Methods and Techniques, LNCS, 2693, 352-373, 2003, Springer-Verlag, Berlin Heidelberg New York , http://dx.doi.org/10.1007/b11721

380. Rosenfeld, S. A. 1997. Bringing business clusters into the mainstream of economic development. *European Planning Studies*, 5(1): 3-23.

381. (RRDTool 2007)
RRDTool, "RRDTool Project Homepage", 2007, http://oss.oetiker.ch/rrdtool/ [Accessed Sept 2007]

382. (Rubin, H. and Natarajan, N. 1994)
Rubin, H. and Natarajan, N., "A distributed software architecture for telecommunication networks", Network, IEEE, 8, [1], 8-17, 1994, 0890-8044, http://dx.doi.org/10.1109/65.260068

383. (Ruffin, M. and Ebert, C. 2004)
Ruffin, M. and Ebert, C., "Using Open Source Software in Product Development: A Primer.", IEEE Software, 21, [1], 82-86, 2004, http://dx.doi.org/10.1109/MS.2004.1259227

384. (Russel, D. R. and English, A. R. 1995)
Russel, D. R. and English, A. R., "Converting the BBC's analogue-based broadcasting network to the digital telecommunications domain", International Broadcasting Convention (IBC 95) 14-18 September 1995, 470-475, 1995, IEE, London , http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=475451

385. (Sartin, Jonathan 2005a)
Sartin, Jonathan, "OpenNMS Tomcat 5.5 JMX HOW-TO", 2005a, OpenNMS, http://www.opennms.org/index.php/Tomcat_5.5_JMX_How-To [Accessed Sept 2007]

386. (Sartin, Jonathan 2005b)
Sartin, Jonathan, "OpenNMS: A Study in Deployment", LinuxPlanet, 2005b, Jupitermedia Corporation, http://www.linuxplanet.com/linuxplanet/reports/5879/1/ [Accessed Sept 2007]

387. (Saunders, M., Lewis, P., & Thornhill, A. 2003)
Saunders, M., Lewis, P., & Thornhill, A. 2003. *Research Methods for Business Students 3rd Edition*. Harlow: Prentice Hall

388. (Saxberg, Borje O. and Slochum, John W. Jr. 1968)
Saxberg, Borje O. and Slochum, John W. Jr., Management Science, 14, [8], B473-B489, 1968, http://links.jstor.org/sici?sici=0025-1909%28196804%2914%3A8%3CB473%3ATMOSM%3E2.0.CO%3B2-Y

389. (SBLIM 2007)
SBLIM, "Standards Based Linux Instrumentation for Manageability (SBLIM)", 2007, http://sblim.wiki.sourceforge.net/ [Accessed Sept 2007]

390. (Schoenwaelder, J 2003)
Schoenwaelder, J, "RFC3535: Overview of the 2002 IAB Network Management Workshop", 2003, IETF, http://www.ietf.org/rfc/rfc3535.txt [Accessed Sept 2007]

391. (Schön, D. A. 1983)
Schön, D. A. 1983. *The reflective practitioner - how professionals think in action*. New York: Basic Books, ISBN.046506874X

392. (Schott, Julie and Westerinen, Andrea 2003)
Schott, Julie and Westerinen, Andrea, "Language of Management Standards,", presentation DMTF Global Management Conference June 16-19 2003 San Jose California, 2003, DMTF, http://www.dmtf.org/events/past/2003/gmc/presentations/GMC03_LanguageofManagementStandards.pdf [Accessed Sept 2007]

393. (SEI 2000)
SEI, "Software Engineering Institute's Capability Maturity Model CMM-SEI, 2000", 2000, http://www.sei.cmu.edu/cmm/ [Accessed Sept 2007]

394. (Seregelyi, E. V. and Kohut, I. A. 1991)
Seregelyi, E. V. and Kohut, I. A., "Managing the R&D-marketing interface", Technology Management : the New International Language, 1991, 153-156, 1991, 0-7803-0161-7, http://dx.doi.org/10.1109/PICMET.1991.183600

395. (Shihada, Basem 2002)
Shihada, Basem, "Conceptual & Concrete Architectures of Open Network Management System (OpenNMS) [Unpublished Phd Course Project CS798 Advanced Topics: Software Architecture. Accessed 10-8-07]", 2002, University of Waterloo, Canada , http://www.cs.uwaterloo.ca/~bshihada/finalproj.pdf

396. (Short, Mike 2006)
Short, Mike, "Mobile TV - Presentation by Mike Short, Vice President-Research & Development, mmO2 plc", Telemanagement World May 15-16 2006, 2006, TMForum, NJ

397. (Shrewsbury, J. Kirk 2001)
Shrewsbury, J. Kirk, "Thresholds, edited by Lawrence Bernstein: Distributed Building Blocks: An IT Foundation for Network and Service Management Software", J.Netw.Syst.Manage., 9, 1-10, 2001, Plenum Press, New York, NY, USA 1064-7570, http://dx.doi.org/10.1023/A:1009405416982

398. (Sidonis 2007)
Sidonis, "Sidonis StateWise Product Homepage", 2007, http://www.sidonis.com/html/products.html [Accessed Sept 2007]

399. (Silva, Nuno, Ryan, Allan, Jorge, Herlander, Reis, Pedro Bettencourt, and Guerra, Maria 2006)
Silva, Nuno, Ryan, Allan, Jorge, Herlander, Reis, Pedro Bettencourt, and Guerra, Maria, "Eurescom Study P1552 Report (D2): Open Source for NGN Management - Technical issues and solutions", 2006, EURESCOM, Heidelberg, Germany , http://www.eurescom.de/activities/studyprogrammes/studyresults/P1500-series/P1552-D2.asp [Accessed Sept 2007]

400. (Silverthorne, Sean 2005)
Silverthorne, Sean, "Microsoft vs. Open Source: Who Will Win? Q&A with:Ramon Casadesus-Masanell and Pankaj Ghemawat", HBS Working Knowledge, 2005, Harvard Business School, http://hbswk.hbs.edu/item/4834.html [Accessed Sept 2007]

401. (Sowa, J and Zachman, John A. 1992)
Sowa, J and Zachman, John A., "Extending and Formalizing the Framework for IS Architecture", IBM Systems Journal, 31, [3], 590-616, 1992, IBM, http://domino.research.ibm.com/tchjr/journalindex.nsf/e90fc5d047c64cbf85256bc80066919c/67e3804651509 4cf85256bfa00685c94?OpenDocument [Accessed Sept 2007]

402. (Spring 2007b)
Spring, "Spring Application Framework Project Homepage", 2007b, http://www.springframework.org/ [Accessed Sept 2007]

403. (Spring 2007a)
Spring, "Spring IDE Project Hompepage", 2007a, http://springide.org/blog/ [Accessed Sept 2007]

404. (Stallman, R. 1999)
Stallman, R. 1999. The GNU Operating System and the Free Software Movement. In C. DiBona, S. Ockman, & M. Stone (Eds.), *Open sources - voices from the open source revolution*: Beijing: O'Reilly,.ISBN 1565925823, http://www.oreilly.com/catalog/opensources/book/stallman.html

405. (Stevenson, Douglas W. 2000)
Stevenson, Douglas W., "MAJI Correlation Engine Specification", 2000, OpenNMS, http://www.opennms.org/maji/Maji-Draft-0.1c.doc [Accessed Sept 2007]

406. (Strang, C J. 2005)
Strang, C J., "Next generation systems architecture - the Matrix", BT Technology Journal, 23, [1], 55-68, 2005, Springer, Netherlands 1573-1995, http://dx.doi.org/10.1007/s10550-005-0107-1

407. (Strassner, John C 2004a)
Strassner, John C, "Autonomic networking - theory and practice", Network Operations and Management Symposium, 2004.NOMS 2004.IEEE/IFIP, 1, 927, 2004a, IEEE 1542-1201

408. (Strassner, John C and Raymer, D. 2006)
Strassner, John C and Raymer, D., "Implementing Next Generation Services Using Policy-Based Management and Autonomic Computing Principles", Network Operations and Management Symposium, 2006.NOMS 2006.10th IEEE/IFIP, 1-15, 2006, IEEE, http://dx.doi.org/10.1109/NOMS.2006.1687611

409. (Strassner, John C. 2004b)
Strassner, John C., "Policy-Based Network Management", Solutions for the Next Generation, 466-470, 2004b, Morgan Kauffmann, San Francisco

410. (Subversion 2007)
Subversion, "Subversion Project Homepage", 2007, http://subversion.tigris.org/ [Accessed Sept 2007]

411. (SUN Microsystems 2006b)
SUN Microsystems, "Java Management Extensions (JMX) 1.4 Specification (Final Release November 9th 2006)", 2006b, http://java.sun.com/javase/6/docs/technotes/guides/jmx/JMX_1_4_specification.pdf [Accessed Sept 2007]

412. (SUN Microsystems 2007)
SUN Microsystems, "Java Foundation Classes : Swing Tutorial", 2007, http://java.sun.com/docs/books/tutorial/uiswing/index.html [Accessed Sept 2007]

413. (SUN Microsystems 2005a)
SUN Microsystems, "Sun Announces Open Source License for Solaris Operating System", 2005a,

Sun Microsystems, http://www.sun.com/smi/Press/sunflash/2005-01/sunflash.20050125.1.xml [Accessed Sept 2007]

414. (SUN Microsystems 2006a)
SUN Microsystems, "Sun Open Sources Java Platform and Releases Source Code Under GPL License Via NetBeans and Java.net Communities", 2006a, Sun Microsystems, http://www.sun.com/smi/Press/sunflash/2006-11/sunflash.20061113.1.xml [Accessed Sept 2007]

415. (SUN Microsystems 2005b)
SUN Microsystems, "Java Management Extensions (JMX)", 2005b, http://java.sun.com/products/JavaManagement/ [Accessed Sept 2007]

416. Susman, G. I. & Evered, R. D. 1978. An Assessment of the Scientific Merits of Action Research. *Administrative Science Quarterly*, 23(4): 582-603.

417. (Szyperski, C., Gruntz, D., & Murer, S. 2003)
Szyperski, C., Gruntz, D., & Murer, S. 2003. *Component software beyond object-oriented programming*. London: Addison-Wesley, ISBN.0201745720 (alk. paper)

418. (Tada, H., Usui, W., and Xu, Jian Wen 2003)
Tada, H., Usui, W., and Xu, Jian Wen, "An approach toward implementation of OSS/BSS using NGOSS", Proceedings of the International Conference on Communication Technology 2003.ICCT 2003.9-11 April 2003, 1, 57-59, 2003, IEEE ISBN: 7-5635-0686-1, http://dx.doi.org/10.1109/ICCT.2003.1209035

419. (TheOpenGroup 2007b)
TheOpenGroup, "OSF-DCE Distributed Computing Environment Homepage", 2007b, http://www.opengroup.org/dce/info/ [Accessed Sept 2007]

420. (TheOpenGroup 2007a)
TheOpenGroup, "Open Pegasus open source DMTF CIM repository", 2007a, The Open Group, http://www.openpegasus.org/ [Accessed Sept 2007]

421. (TheOpenGroup 2007c)
TheOpenGroup, "The Open Group Homepage", 2007c, http://www.opengroup.org [Accessed Sept 2007]

422. (Thompson, A. A. & Strickland, A. J. 2003)
Thompson, A. A. & Strickland, A. J. 2003. *Strategic Management - concepts and cases*. Boston: McGraw-Hill/Irwin, ISBN.0072443715

423. (Thorogood, Alan and Yetton, Phillip 2004)
Thorogood, Alan and Yetton, Phillip, "Reducing the technical complexity and business risk of major systems projects", Proceedings of the 37th Annual Hawaii International Conference on System Sciences 2004, Track 8 Volume 8, 80253.2, 2004, IEEE Computer Society, Washington, DC, USA 0-7695-2056-1, http://dx.doi.org/10.1109/HICSS.2004.1265608

424. (TigerStripeTigerStripe, "Tiger Stripe Development Community", http://www.tigerstripedev.net/ [Accessed Sept 2007]

425. (TINA-C 2007)
TINA-C, "Telecommunications Information Networking Architecture Consortium(TINA-C) Homepage", 2007, http://www.tinac.com [Accessed Sept 2007]

426. (TMForum 2006a)
TMForum, "GB929 Telecom Application Map - The OSS BSS Systems Landscape Release 2", 2006a, Telemanagement Forum, NJ , http://www.tmforum.org/page32370.aspx [Accessed Sept 2007]

427. (TMForum 2006b)
TMForum, "NGOSS Contract Management Program", 2006b, Telemanagement Forum, NJ , http://www.tmforum.org/browse.aspx?catID=3951&linkID=32150 [Accessed Sept 2007]

428. (TMForum 2005e)
TMForum, "TMF517 Multi Technology Operations Systems Interface (MTOSI) Business Agreement", 2005e, Telemanagement Forum, NJ , http://www.tmforum.org/browse.aspx?catID=4133&linkID=32481

429. (TMForum 2007b)
TMForum, "GB921 R7-0 eTOM Solution Suite (GB921) Release 7.0", 2007b, TMForum, NJ , http://www.tmforum.org/browse.aspx?catID=1649 [Accessed Sept 2007]

430. (TMForum 1989)
TMForum, "Telemanagement Forum IPR Guide and IPR Agreement", 1989, http://www.tmforum.org/browse.aspx?catID=2211&linkID=29757&docID=3068 [Accessed Sept 2007]

431. (TMForum 2007a)
TMForum, "TMForum Catalyst Program Homepage", 2007a, http://www.tmforum.org/pages/786/default.aspx [Accessed Sept 2007]

432. (TMForum 2004c)
TMForum, "Open OSS - Project Charter", The 1st step in an alternative approach to drive the implementation of NGOSS, 2004c, www.tmforum.org

433. (TMForum 2007c)
TMForum, "Milestones in the history of the TMForum", 2007c, http://www.tmforum.org/browse.aspx?catID=1086 [Accessed Sept 2007]

434. (TMForum 2006c)

TMForum, "TMF053 The NGOSS Technology Neutral Architecture Version 5.7", NGOSS Architecture Solution Suite Release 6.0 January 01, 2006, 2006c, TMForum, NJ , http://www.tmforum.org/page31905.aspx

435. (TMForum 2004i)
TMForum, "TMF053b: NGOSS Architecture Technology Neutral Specification Contract Description: Business and System Views", NGOSS Architecture Solution Suite Release 6.0 January 01, 2006, 2004i, http://www.tmforum.org/page31905.aspx [Accessed Sept 2007]

436. (TMForum 2007g)
TMForum, "TMForum Prosspero Program Homepage", 2007g, http://www.prosspero.org/ [Accessed Sept 2007]

437. (TMForum 2004a)
TMForum, "GB927 v1.1 NGOSS Lifecycle Methodology", 2004a, http://www.tmforum.org/browse.aspx?catID=2025&linkID=29276 [Accessed Sept 2007]

438. (TMForum 2005b)
TMForum, "GB932 DMTF/TMF Model Alignment for the Physical Sub-model R1-0, V0.1.14", CIM-SID Solution Suite Release 6.0, 2005b, Telemanagement Forum, NJ , http://www.tmforum.org/browse.aspx?catID=3026&linkID=31908 [Accessed Sept 2007]

439. (TMForum 2005d)
TMForum, "GB933 DMTF/TMF Model Alignment for the SID Logical Resources and CIM Networks Sub-models R6-0, V0.6", CIM-SID Solution Suite Release 6.0, 2005d, Telemanagement Forum, NJ , http://www.tmforum.org/browse.aspx?catID=3026&linkID=31908 [Accessed Sept 2007]

440. (TMForum 2004f)
TMForum, "GB922 NGOSS Shared Information/Data Model (SID) Release 4.5", 2004f, TMForum, NJ , http://www.tmforum.org/browse.aspx?catID=2385&linkID=31149 [Accessed Sept 2007]

441. (TMForum 2004h)
TMForum, "OpenOSS Catalyst 2005 Project Charter - The 1st step in an alternative approach to drive the implementation of NGOSS", 2004h, http://www.tmforum.org/Nice2005/OpenOSS/29972/article.html [Accessed Sept 2007]

442. (TMForum 2001)
TMForum, "TMF GB-909, Generic Requirements for Telecommunications Management Building Blocks, version 2.1, January 2001", 2001, Telemanagement Forum, NJ

443. (TMForum 2004e)
TMForum, "GB917 SLA Management Handbook Version 2.0", 2004e, Telemanagement Forum, NJ , http://www.tmforum.org/DocumentLibrary/SLAHandbookSolution/2035/Home.html [Accessed Sept 2007]

444. (TMForum 2004g)
TMForum, "GB922 Shared Information and Data (SID) Model - Addendum-4SO Service Overview Version 2.1", Contained in NGOSS Release 4.5 SID Model Suite (Phase 5), 2004g, Telemanagement Forum, NJ , http://www.tmforum.org/browse.aspx?catID=2024&linkID=30024 [Accessed Sept 2007]

445. (TMForum 2005f)
TMForum, "TMF517 Multi Technology Operations Systems Interface (MTOSI) Business Agreement Release 1.1", 2005f, Telemanagement Forum, NJ , http://www.tmforum.org/browse.aspx?catID=4133&linkID=32481 [Accessed Sept 2007]

446. (TMForum 2006d)
TMForum, "TMFC5171 SID and Other Models (SID & MTMN) (Presentation)", 2006d, http://www.tmforum.org/browse.aspx?catID=3057&linkID=32044&docID=5171 [Accessed Sept 2007]

447. (TMForum 2007d)
TMForum, "Telemanagement Forum Open OSS Program Team Homepage", 2007d, http://www.tmforum.org/browse.aspx?catID=2602&linkID=31021 [Accessed Sept 2007]

448. (TMForum 2007e)
TMForum, "Telemanagement Forum University Program Homepage", 2007e, http://www.tmforum.org/browse.aspx?catID=2690&linkID=31338 [Accessed Sept 2007]

449. (TMForum 2004b)
TMForum, "NGOSS Overview", 2004b, http://www.tmforum.org/browse.asp?catID=1912 [Accessed Sept 2007]

450. (TMForum 2004d)
TMForum, "OSS through Java as an Implementation of NGOSS A White Paper", 2004d, http://www.tmforum.org/browse.asp?catID=1923&linkID=29240 [Accessed Sept 2007]

451. (TMForum 2007f)
TMForum, "Telemanagement Forum, A New Jersey Non Profit Corporation; Address: 240 Headquarters Plaza, East Tower, 10th Floor, Morristown, NJ 07960-6628, USA", 2007f, www.tmforum.org [Accessed Sept 2007]

452. (TMForum 2003)
TMForum, "NGOSS Architecture Technology Neutral Specification Metamodel - TMF053D, Member Evaluation Version 1.0, February 2003", 2003, http://www.tmforum.org/page28001.aspx [Accessed Sept 2007]

453. (TMForum 2005a)
TMForum, "GB932 DMTF/TMF Model Alignment for the Physical Sub-model", NGOSS Release 5.0 Solution Suite, 2005a, Telemanagement Forum, NJ

454. (TMForum 2005c)
TMForum, "GB933 DMTF/TMF Model Alignment for the SID Logical Resources and CIM Networks Sub-models", NGOSS Release 5.0 Solution Suite, 2005c, Telemanagement Forum, NJ

455. (Tolkien, J. R. R. 1954)
Tolkien, J. R. R. 1954. *The Lord of the Rings*. London: Allen & Unwin

456. (Torchiano, M. and Morisio, M. 2004)
Torchiano, M. and Morisio, M., "Overlooked aspects of COTS-based development", Software, IEEE, 21, [2], 88-93, 2004, IEEE 0740-7459, http://dx.doi.org/10.1109/MS.2004.1270770

457. (Trigeorgis, L. 1996)
Trigeorgis, L. 1996. *Real options managerial flexibility and strategy in resource allocation.* Cambridge, Mass: MIT Press, ISBN.026220102X

458. (Trigger, J and Harrison, M 1-1-2006)
Trigger, J and Harrison, M, "Six steps to excellent customer service", BT Technology Journal, 24, [1], 117-126, 1-1-2006, Springer, Netherlands 1573-1995, http://dx.doi.org/10.1007/s10550-006-0027-8

459. (Truphone 2007)
Truphone, "Truphone Technical Information", 2007, Software Cellular Network Ltd, http://www.truphone.com/more_info/tech.tru [Accessed Sept 2007]

460. (UKAIS 2007)
UKAIS, "UK Academy for Information Systems (UKAIS) Definition of Information Systems", 2007, http://www.turningcourse.com/ukais/isdefn.pdf [Accessed Sept 2007]

461. (Vaarandi, Risto 2002a)
Vaarandi, Risto, "Platform independent event correlation tool for network management", IEEE/IFIP Network Operations and Management Symposium, 2002.NOMS 2002.2002 I5-19 April 2002, 907-909, 2002a, IEEE, http://dx.doi.org/10.1109/NOMS.2002.1015640

462. (Vaarandi, Risto 2002b)
Vaarandi, Risto, "SEC - a lightweight event correlation tool", IEEE Workshop on IP Operations and Management, 2002, 111-115, 2002b, IEEE, http://dx.doi.org/10.1109/IPOM.2002.1045765

463. (van de Kar , Elizabeth A. M. 2005)
van de Kar , Elizabeth A. M., "The design of a mobile information and entertainment service on a UMTS testbed", ICEC '05: Proceedings of the 7th international conference on Electronic commerce, 607-614, 2005, ACM Press, New York, NY, USA 1-59593-112-0, http://doi.acm.org/10.1145/1089551.1089660 ,

464. (Vessey, Iris, Ramesh, V, and Glass, Robert L. 2001)
Vessey, Iris, Ramesh, V, and Glass, Robert L., "A unified classification system for research in the computing disciplines. Technical Report TR107-1", 2001, Kelley School of Business, Indiana University, Indiana , http://www.kelley.iu.edu/ardennis/wp/tr107-1.doc [Accessed Sept 2007]

465. (Vitharana, Padmal 2003)
Vitharana, Padmal, "Risks and Challenges of Component-Based Software Development", Communications of the ACM, 46, [8], 67-72, 2003, http://doi.acm.org/10.1145/859670.859671

466. (Voas, J. 1998)
Voas, J., "COTS Software – the Economical Choice?", IEEE Software, 15, [2], 16-19, 1998, http://dx.doi.org/10.1109/52.663777

467. (von Hippel, E. 2005)
von Hippel, E. 2005. Open Source Software Projects as "User Innovation Networks". In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*: 267-278. Cambridge, Mass: MIT Press,.ISBN 0-262-06246-1, http://mitpress.mit.edu/books/chapters/0262062461chap14.pdf

468. (von Hippel, Eric 1994)
von Hippel, Eric, "'Sticky Information', and the Locus of Problem Solving: Implications for Innovation", Management Science, 40, [4], 429-439, 1994, http://links.jstor.org/sici?sici=0025-1909(199404)40%3A4%3C429%3A%22IATLO%3E2.0.CO%3B2-M

469. (von Hippel, Eric 2001b)
von Hippel, Eric, "User toolkits for innovation", Journal of Product Innovation Management, 18, [4], 247-257, 2001b, Blackwell, http://dx.doi.org/10.1111/1540-5885.1840247

470. (von Hippel, Eric 2001a)
von Hippel, Eric, "Innovation by user communities: Learning from open source software", Sloan Management Review, 42, [4], 82-86, 2001a, MIT Sloan School of Management, Cambridge, MA , http://sloanreview.mit.edu/smr/issue/2001/summer/8/

471. (von Hippel, Eric and Tyre, Marcie J 1995)
von Hippel, Eric and Tyre, Marcie J, "How 'Learning by Doing' is Done: Problem Identification in Novel Process Equipment.", Research Policy, 24, [1], 1-12, 1995, Elsevier Science B.V. 0048-7333, http://dx.doi.org/10.1016/0048-7333(93)00747-H

472. (von Krogh, Georg and von Hippel, Eric 2006)
von Krogh, Georg and von Hippel, Eric, "The Promise of Research on Open Source Software", Management Science, 52, [7], 975-983, 2006, http://dx.doi.org/10.1287/mnsc.1060.0560

473. (W3C 2003)
W3C, "Scalable Vector Graphics (SVG) 1.1 Specification W3C Recommendation 14 January 2003", 2003, http://www.w3.org/TR/SVG11/ [Accessed Sept 2007]

474. (W3C 2004)
W3C, "Web Services Architecture - W3C Working Group Note (11 February 2004)", 2004, World Wide Web Consortium, http://www.w3.org/TR/ws-arch/#whatis

475. (W3C 2007)
W3C, "World Wide Web Consortium Homepage", 2007, http://www.w3.org/ [Accessed Sept 2007]

476. (Wallnau, Kurt, Bachmann, Felix, Bass, Len, Buhman, Charles, Comella-Dorda, Santiago, Long, Fred, Robert, John, and Seacord, Robert 2000)
Wallnau, Kurt, Bachmann, Felix, Bass, Len, Buhman, Charles, Comella-Dorda, Santiago, Long, Fred, Robert, John, and Seacord, Robert, "Technical Concepts of Component-Based Software Engineering,Volume II: 2nd Edition, TECHNICAL REPORT CMU/SEI-2000-TR-008 ESC-TR-2000-007", 2000, Carnegie Mellon University, Software Engineering Institute, Pittsburg PA , http://www.sei.cmu.edu/pub/documents/00.reports/pdf/00tr008.pdf [Accessed Sept 2007]

477. (Ward, Stephen and Chapman, Chris B. 2003)
Ward, Stephen and Chapman, Chris B., "Transforming project risk management into project uncertainty management", International Journal of Project Management, 21, [2], 97-105, 2003, http://dx.doi.org/10.1016/S0263-7863(01)00080-1

478. (Wardlaw, M. 2005)
Wardlaw, M., "Intelligence and mobility for BT's next generation networks", BT Technology Journal, 23, [1], 28-47, 2005, Springer, Netherlands 1573-1995, http://dx.doi.org/10.1007/s10550-005-0105-3

479. (WBEMservices 2006)
WBEMservices, "WBEMservices", 2006, http://wbemservices.sourceforge.net/ [Accessed Sept 2007]

480. (Weinstock, C. B. & Hissam, S. A. 2005)
Weinstock, C. B. & Hissam, S. A. 2005. Making Lightning Strike Twice. In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*: 144-159. Cambridge, Mass: MIT Press,.ISBN 0-262-06246-1

481. (West, Joel 2004)
West, Joel, "What are Open Standards? Implications for Adoption, Competition and Policy", Standards and Public Policy conference May 11th 2004, 2004, Federal Reserve Bank of Chicago, Chicago, Illinois , http://www.chicagofed.org/news_and_conferences/conferences_and_events/files/west.pdf [Accessed Sept 2007]

482. (Wikipedea 2007)
Wikipedea, "Marchitecture", 2007, http://en.wikipedia.org/wiki/Marchitecture [Accessed Sept 2007]

483. (Wikipedia 2007)
Wikipedia, "Truphone Description and History (Wikipedia Entry)", 2007, http://en.wikipedia.org/wiki/Truphone [Accessed Sept 2007]

484. (Winer, Dave 1999)
Winer, Dave, "XML-RPC Specification Update 3 (Tue, Jun 15, 1999)", 1999, http://www.xmlrpc.com/spec#update3 [Accessed Sept 2007]

485. (Winslow, Maria 2004)
Winslow, Maria, "Huge Savings with OpenNMS for Managed Hosting Company", Linux, 2004, SYS-CON Media, http://linux.sys-con.com/read/46193.htm [Accessed Sept 2007]

486. (Worthington, David 2005)
Worthington, David, "IBM turns to open source development", BetaNews, 2005, BetaNews Inc., http://www.betanews.com/article/IBM_Turns_to_Open_Source_Development/1118688437/1 [Accessed Sept 2007]

487. (XMLBeans 2006)
XMLBeans, "XMLBeans", 2006, http://xmlbeans.apache.org/ [Accessed Sept 2007]

488. (Yang, K, Ou, S, Azmoodeh, M., and Georgalas, N 2005)
Yang, K, Ou, S, Azmoodeh, M., and Georgalas, N, "Policy-based model-driven engineering of pervasive services and the associated OSS", BT Technology Journal, 23, [3], 162-174, 2005, Springer, Netherlands 1573-1995, http://dx.doi.org/10.1007/s10550-005-0039-9

489. (Zachman, John A. 1987)
Zachman, John A., "A framework for information systems architecture", IBM Systems Journal, 26, [3], 276-292, 1987, IBM, http://domino.research.ibm.com/tchjr/journalindex.nsf/e90fc5d047e64ebf85256bc80066919c/d9d379f60c3859fd85256bfa00685bd7?OpenDocument [Accessed Sept 2007]

Glossary

Glossary

Term	Description
Ajax	Asynchronous JavaScript and XML
ASI	DVB Asynchronous Serial Interface (*ETSI 2006c*)
B2B	Business to Business (e-commerce)
Bellcore	Bellcore was a consortium established by the Regional Bell Operating Companies upon their separation from AT&T. now Telcordia Technologies
BOF	Birds Of a Feather session (TMForum)
BSS	Business Support Systems
CAPEX	Capital Expenditure
Catalyst	TMForum program for demonstrating NGOSS interoperability
CFO	Chief Financial Officer
CIM	DMTF's Common Information Model (*DMTF 2006*)
CIO	Chief Information Officer
CLI	Command Line Interface
CMIP/CMIS	Common Management Information Protocol / Service
CMOT	CMip Over Tcp/ip defined in RFC 1095 (*IETF 1989*)
CORBA	Common Object Request Broker Architecture
CRM	Customer Relationship Management
CRM	Customer Relationship Management
CS	Computer Science
CSV	Comma Separated Variable
CSV	Comma Separated Variable
CTO	Chief Technical Officer
CTP	Connection Termination Point
DAB	Digital Audio Broadcasting (*ETSI 2006b*)
DMTF	Distributed Management Task Force (*DMTF 2007a*)
DVB-H	Digital Video Broadcasting - Handheld (*DVB-H 2007*)
DVB-S	Digital Video Broadcasting - Satellite (*ETSI 1998a*)
DVB-T	Digital Video Broadcasting - Terrestrial (*ETSI 2004*)
ebXML	Electronic Business using eXtensible Markup Language (*ebXML 2007*)
EJB	Enterprise Java Bean
EJB3	Next Generation of Enterprise Java Bean after J2EE
ETSI	European Telecommunications Standards Institute
FCAPS	Fault, Configuration, Accounting, Performance and Security
FlowThru	EU ACTS project AC335 (*FlowThrough 2000a*) (March 1998 - Feb 2000)
FORM	EU IST 5th framework project FORM (April 2000 - May 2002) (*FORM 2002*)
FRAND	Fair, Reasonable, And Non-Discriminatory IPR policy - see also RF (royalty free)
Freeview	UK DVB-T service (*Freeview 2007*)
GDMO	Guidelines for the Definition of Managed Objects
ILEC	Incumbent Local Exchange Carrier (USA)
INA	Information Networking Architecture
IPR	Intellectual property rights
IRC	Internet RElay Chat
IRTF	Internet Research Task Force
IS	Information Systems
ITIL	Information Technology Infrastructure Library (*ITIL 2007*)
ITU-T	International Telecommunication Union Telecommunication Standardization Sector
J2EE	Jave 2 Enterprise Edition
JAR	Java ARchive file
JCP	Java Community Process (*JCP 2004a*)
JDBC	Java Database Connectivity API
JSP	Java Server Pages
JSR	Java Specification Requests
JVM	Java Virtual Machine
JVT	OSS/J Java Value Type pattern
KPI	Key performance Indicator
KQI	Key Quality Indicator
Liberty Alliance	Open Standards Organisation for federated identity management (*LibertyAlliance 2007*)
MDA	Model Driven Architecture (*OMG 2007a*)
MIB	Management Information Base
MPLS	Multi-Protocol Label Switching
NDA	Non Disclosure Agreement

Term	Description
NetSNMP	open source SNMP library written in C
NgnAS	Agilent's Next Generation Network Analysis System (*Agilent 2002*)
NGOSS	New Generation Operations Systems and Software (*TMForum 2004b*)
NGOSSoil	NGOSS Object Integration Layer - Open Source project proposed by Nortel
NMRG	IRTF Network Management Research Group
NPI	New Product Introduction
NPV	Net Present Value
OAM	Operation, Administration, and Maintenance
OASIS	Organization for the Advancement of Structured Information Standards
OGP	Order of the Green Polo' (OpenNMS developers)
OMG	Object Management Group
OMNIPoint	Early TMForum standard promoting OSI integration
Open OSS	Open Source program initiated in the TMForum (*TMForum 2007d*)
OpenNMS	Open source Network Management Project (*OpenNMS 2007c*)
OpenOSS	Open Source Operational Support System (*OpenOSS 2007*)
OPEX	Operational Expenditure
OSCA	Operations Systems Computing Architecture (*Mills, J. A. et al. 1990*)
OSF/DCE	Open Software Foundation's Distributed Computing Environment (*TheOpenGroup 2007b*)
OSGi	Open Services Gateway initiative (*OSGi 2007*)
OSI	Open Systems Interconnection model
OSS	Operational Support Systems
OSS/J	OSS through Java (*OSS/J 2007b*)
PAT	Program Association Table (DVB)
PES	Packet Elemental Stream (DVB)
PID	Packet Stream Identifier (DVB)
Prosspero	TMForum Interface Productisation Program (*TMForum 2007g*)
PTT	Postal Telephone and Telegraph, a government agency in many countries worldwide
Qos	Quality of Service
qosd	Component in OpenNMS qosdaemon which publishes an alarm list
qosdaemon	OpenNMS daemon which implements the OSS/J FM interface
qosdrx	Component in qosdaemon which receives OSS/J alarms
Real Option	A real option is the right, but not the obligation, to undertake some business decision,
RF	Radio Frequency
RF	Royalty Free (RF) IPR policy - see also FRAND
RI	Reference Implementation
SANRR	Scope, Analyse, Normalise, Rationalise, Rectify (NGOSS methodology)
SAP	Service Access Point
SAP	Service Access Point (SLA)
SCADA	Supervisory Control And Data Acquisition
SDH	Synchronous Digital Hierarchy (SDH) - optical communication system - like SONET
SDO	Standards Development organisations
SDO	Standards Development Organisation
SE	Software Engineering
servlet	A Java technology used to render web pages
SI	Systems Integrator
SID	Shared Information/Data (SID) model
SLA	Service Level Agreement
SME	Small and Medium Enterprises (< 250 employees in EU)
SMIng	Next Generation Structure of Management Information (*NMRG 2003*)
SNMP	Simple Network Management Protocol (*IETF 1990*)
SONET	Synchronous Optical Network - optical transmission system like SDH
SPLC	TMForum Service Provider Leadership Council
StateWise	Correlation product from Sidonis (*Sidonis 2007*)
SVG	Scalable Vector Graphics (*W3C 2003*)
TCK	Technology Compatibility Kit
TeMIP	Telecommunications Management Information Platform (HP product) (*HP 2007a*)
TINA-C	Telecommunications Information Networking Architecture Consortium (*TINA-C 2007*)
TL1	Transaction Language 1
TMForum	Telemanagement Forum (*TMForum 2007f*)
TMN	Telecommunications Management Network (*ITU-T 2000b*)
TMW	Telemanagement World (trade conference)
Transposer	In broadcasting a relay which retransmits a signal on a different frequency without demodulation
TSA	Transport Stream Analyser

Term	Description
UHF PAL	Analogue colour TV transmission standard
UoS	University of Southampton
VLAN	Virtual LAN
VoIP	Voice over Internet Protocol
W3C	The World Wide Web Consortium (*W3C 2007*)
WfMC	Workflow Management Coalition
XML	eXtensible Markup Language
XMLBeans	Open source XML marshalling framework (*XMLBeans 2006*)
XP	eXtreme Programming an agile software development methodology
XSLT	EXtensible Stylesheet Language Transformations
XVT	OSS/J XML Value Type pattern





University
of Southampton

Improving the practice of Operations Support Systems in the Telecommunications Industry using Open Source

Appendices

by

Craig Gallen

A Thesis submitted for the degree of
Doctor of Engineering

School of Electronics and Computer Science,
University of Southampton,
United Kingdom.

Date: November 2007

Appendices Table of Contents

Appendix A	Chronology of the Action Research Program	226
Appendix B	Meeting Invocom's Research Objectives	238
Appendix C	Impact of Open Source on eTOM Processes	246
Appendix D	Open OSS Stakeholder Expectations	252
Appendix E	OpenNMS Case Study	266
Appendix F	OpenNMS OSS/J interface Design	289
Appendix G	OpenOSS Catalyst Final Technical Report	310

Appendix A - Chronology of the Action Research Program

Appendix A Chronology of the Action Research Program

A.1 A Concise Project History

The overall research program has been partitioned into 4 key projects and each project has its own cycle of action research proceeding in parallel with the others.

1. Initial OpenOSS Catalyst Project
2. Follow on activities from OpenOSS Catalyst Project
3. Nortel contributed NGOSSoil project
4. Arqiva DVB-T digital TV management project

The open source development of the OpenNMS OSS/J capability has proceeded throughout all of these projects and so is not broken out into a separate project but thought of as an enabler for the projects described above.

The purpose of the following history is to set the context in which the Action Research has been done in the TMForum and through which qualitative data has been obtained and analysed through reflective learning. It is not the intention (and would not be possible) to provide a blow by blow account of all of the activities which ensued. However the selected key events and project documentation are intended to afford a degree of 'recoverability' (*Checkland, P. et al. 1998*) to the research process. In the following account the numbers in brackets [n] refer to the business events listed in the table in Appendix A.2.

The Initial OpenOSS Catalyst Project ❶ commenced with a Birds of a Feather (BOF) Session at TMW Nice 2004. Invocom's prior agreement to support the BOF had followed several months of market diagnosis and action planning [1] (see Appendix B). This BOF [2] generated considerable interest particularly from BT who along with Invocom sponsored a technical investigation using undergraduate students at the University of Southampton during the summer of 2004 [7]. In parallel BT recruited a full time employee (Gary Bruce) to run their Open Source program.

Other contacts from the BOF, including Sun Microsystems [6], were followed up and invited to support a further BOF at TMW Long Beach in October of 2004 which proposed an open source catalyst project (OpenOSS) [11]. The OpenOSS BOF and a demonstration of the students' summer work raised the profile of open source at TMW Long Beach and generated a list of contacts who were invited to a two day workshop at the University of Southampton in December 2004 [12]. The workshop agreed a set of objectives for the OpenOSS Catalyst project which a number (but not all) of the attendees agreed to resource. The workshop also brokered a link between the open source catalyst and a NGOSS Model Driven Architecture catalyst such that the OpenOSS catalyst would be modelled by the MDA catalyst in UML as NGOSS contracts.

Agilent offered a major contribution to the OpenOSS Catalyst in the form of their NgnAS VoIP management product (*Agilent 2002*). The technical architecture for the Catalyst test environment was designed by the researcher, Craig Gallen with Agilent lead architect Doug Carson and Invocom network architects during December 2004 and early January 2005 [13]. The overall plan was finalised during face to face planning meetings over 5 days at the TMForum Team Action Week in January 2005 [16].

The OpenOSS Catalyst open source development was partitioned between Agilent, a team at Budapest Institute of Technology and Business (BUTE) under Tien Van Do and a team at the University of Southampton supported by a developer, Paul Robson from BT. Integration was to be performed by Cognizant technologies in India, project management by Paulo Imbalzano of Invocom and marketing by Peter Mottishaw of Agilent. Leadership of the project was shared between Gary Bruce (BT) and Craig Gallen.

The staging of the OpenOSS Catalyst at TMW Nice 2005 represented a significant technical, logistical and marketing achievement. The Catalyst itself was very well received and generated a useful mailing list of parties interested following open source activities in the TMForum [23]. The full technical report from the project is included in Appendix G.

The project labelled 'Follow on activities from OpenOSS Catalyst Project' ❷ is a catch all for a number of options which were explored to maintain the momentum of OpenOSS after the Catalyst. A number of executive level meetings including service providers COLT, Vodafone, BT and C&W were convened to explore how the project could move forwards [20,21,23,29] but none of them generated a clear direction. BUTE, FOCUS and the University of Southampton proposed bidding for EU IST FW6 funding to properly resource a project but none of the industrial partners wanted to put effort into a bid [20].

To maintain technical momentum BT employed a student over the summer of 2005 to investigate an open source CRM System, which resulted in a paper (*Bruce, G. L. et al. 2006*) [27]. Agilent Technologies and the University of Southampton employed summer students to further the OSS/J work with OpenNMS and investigated the OSS/J inventory interface [28]. This work was presented at JBoss World in October 2005 [32] (*Gallen, C. R. et al. 2005a*).

In the meantime an initial contact from Nortel during TMW Nice was pursued by Craig Gallen through the summer of 2005. Nortel wanted to investigate the possible business benefits of 'open sourcing' a considerable body of java code which formed part of an existing network management platform. Various meetings were brokered with the executive sponsors of the OpenOSS team[24,29,33] which eventually led to Nortel proposing what became known as the NGOSSoil project ❸. This proposal was launched at a BOF at TMW Dallas 2005 [35] and subsequently fleshed out at Team Action Week January 2006 [38].

Nortel's precondition for releasing the code with resources to staff a project was demonstrable interest from some service providers. Over several months Gary Bruce (BT) and Graham Calvert (a Nortel Director of OSS marketing) lobbied the TMForum members extensively to gain support for the project. However despite having 5 vendors signed up to support the project, no service provider was willing to contribute resources and Nortel reluctantly withdrew the offer to the TMForum in April 2006 [42].

In parallel with all of the above TMForum activities, the researcher, Craig Gallen had also been pursuing another opportunity to use open source in an NGOSS proof of concept, this time outside the telecoms industry and in the world of broadcasting. The TMForum was making moves to establish a converged content management work stream and the opportunity to draw in a

Broadcaster to the TMForum's work was very appealing. A very early initial contact in August 2004 lead [9] to renewed interest from Arqiva in August 2005 [30,31]. This grew into the Arqiva DVB-T digital TV management project [4] which is discussed in Chapter 5.

Since both the OpenOSS Catalyst and the DVB-T work depended on extending an open source project, OpenNMS, the work with OpenNMS became a study in itself which provided insights into the practical issues of working with an open source OSS project. This is discussed further in Chapter 6.

A.2 Detailed project chronology

The following table identifies the various phases of action research according to the 5 stages proposed by (*Susman, G. I. et al. 1978*) (Table 6, Section 2.1) against a chronology of key business events during the 3 year research program. Where possible references to key project documents (presentations, papers, minutes and emails) which have been used in the analysis are provided.

<table>
<tr><th></th><th>From</th><th>To</th><th>Project</th><th>Intervention</th><th>AR Phase</th><th>Details</th></tr>
<tr><td>1.</td><td>Sept 2003</td><td>March 2004</td><td>1</td><td>Initial investigations with Invocom</td><td>1) diagnosing:
2) action planning;</td><td>1) diagnosing:
broad market and technical research setting initial objectives and research plan (Gallen, C. R. 2003a) , (Gallen, C. R. 2003b)
2) action planning;
Working with Invocom and TM Forum to agree form and content of Birds of a Feather session</td></tr>
<tr><td>2.</td><td>1/3/2004</td><td></td><td>1</td><td>Call for Participation in new TM Forum Technical Projects</td><td>3) action taking;
4) evaluating</td><td>3) action taking
Published intention to hold a Birds of a Feather session as a call for participation in an Open Source NGOSS project
BiMonthly eNewsletter: March 2004, No 56 http://www.tmforum.org/newscentral/issue_56.html
4) evaluating
Received contacts from
<table><tr><td>BT Exact</td><td>Dr Manooch Azmoodeh
Future ICT Management Systems
Interested in looking at using OMG's MDA with NGOSS</td></tr><tr><td>SUN Microsystems</td><td>Christophe Ebro
OSS SW architect in OSS/J team
Interested in Open Source development, JMX, OSS/J extensions for MTNM</td></tr><tr><td>QinetiQ</td><td>Helena Kelly
Interested in working with Southampton University on possible NGOSS prototype</td></tr><tr><td>AT&T Labs,</td><td>Jenny Huang
OSS Architecture & Strategic Standards
Interested in using Oasis XRI/XDI to implement SID (an ontology of Telecom domain)</td></tr><tr><td>Instituto de Telecomunicações Portugal</td><td>Nuno Silva
Doing work with Open Source in Eurescom</td></tr></table></td></tr>
<tr><td>3.</td><td>6/5/2004</td><td></td><td>1</td><td>Initial meetings with BT at Martlesham;</td><td>4) evaluating
2) action planning;</td><td>4) evaluating
Dr Manooch Azmoodeh, John Wittegreffe, N. Georgalas,
Reviewed BT's interest in open source / OSS/J and MDA (Georgalas, N. & Azmoodeh, M. 2004)
2) action planning
Agreed joint BT, Invocom messages for BOF</td></tr>
<tr><td>4.</td><td>17/05/2004</td><td>21/05/2004</td><td>1</td><td>Telemanagement World Nice</td><td>3) action taking;
4) evaluating</td><td>3) action taking;
Host 'Birds of a Feather' session – 'Open source solutions for NGOSS'
4) evaluating
Significant interest - 19 attendees at BOF
Detailed conversations with AT&T & SUN Microsystems and TM Forum staff</td></tr>
</table>

	From	To	Project	Intervention	AR Phase	Details
5.	24/05/2004	01/02/2005	1	Open OSS project Development market development Technical feasibility study	2) action planning;	4) evaluating • Contact all BOF attendees and gauge use of open source and interest in participating in project 2) action planning; • Agree objectives for BT and Invocom to sponsor 3 summer students to do Open Source Technical feasibility study • Evaluate available Open Source components and plan summer project
6.	23/6/2004		1	Meeting with Christophe Ebro, SUN Grenoble, France	4) evaluating	4) evaluating • Detailed face to face meeting with Christophe Ebro, SUN OSS/J Manager. Review of status of OSS/J and possible role of Open Source 2) action planning; • Christophe Ebro agrees to support students in using OSS/J over the summer.
7.	30/08/2004	20/09/2004	1	Student Summer Project	3) action taking; 4) evaluating	3) action taking; • BT and Invocom sponsor 3 students over summer. • Develop OSS/J TT interface between OpenNMS www.openmms.org and Open source Ticket Request System (OTRS) http://otrs.org/ 4) evaluating • Written report produced on coding experience. Feedback provided to SUN OSS/J team
8.	08/2004	10/2004	1	BT, Sun, Invocom and UoS plan TM Forum Catalyst project; OpenOSS - Public, cross-industry project proving Open Source management solutions	2) action planning;	2) action planning; • British Telecom employ Gary Bruce to run the Open Source program for BT • Weekly teleconferences arranged to plan for an open call to launch a TMForum Catalyst project at TMW in November
9.	11/8/2004		4	First meeting with Bob Potter NTL (Arqiva)	1) diagnosing	1) diagnosing • Presented Open OSS project brief to Bob Potter of NTL (later spun off as Arqiva). This was an exploratory meeting to see if there was any interest in applying work to Broadcast systems • (slides: Open OSS Project briefing for NTL 11-08-04.ppt)
10.	1/10/2004		1	Open OSS discussion with T-Systems 1 October 2004	4) evaluating	4) evaluating • Face to face meeting with Steffen Hartmann, T-Systems Nuremburg to understand their interest in Open source and whether they would like to get involved in OpenOSS. Found they had developed significant internal capability using open source. • (T-Systems Minutes 0.2 27-10-04.doc)
11.	11/10/2004	14/10/2004	1	Telemanagement World Long Beach 2004 BOF Propose Open OSS Catalyst	3) action taking; 4) evaluating 5) specifying learning	3) action taking; • Ran 'Birds of a Feather' session as open call for involvement in TM Forum Catalyst project; OpenOSS 4) evaluating • Significant interest generated and OpenOSS Email list created by TMForum staff. • Open Source mentioned in keynote speech as a significant development 5) specifying learning • Students presented their open source work with OpenNMS and discussed issues in using open source • Dave Hustace form OpenNMS attended and evaluated the project

	From	To	Project	Intervention	AR Phase	Details
12.	02/12/2004	03/12/2004	1	Open OSS workshop – University of Southampton	1) diagnosing 2) action planning;	3) action planning; • 10 companies participate in joint Open OSS and NGOSS/MDA planning sessions at Southampton University over 2 days. • Agreed basic objectives and plan for a catalyst. Gary Bruce agreed to draft project Charter and sign up willing participants • (Open OSS Workshop Minutes;2b 8-12-04.doc)
13.	21/12/2004		1	Meetings with Agilent Scotland Peter Mottishaw / Doug Carson	2) action planning;	3) action planning; • Met in Scotland with Agilent architect Doug Carson and marketing manager Peter Mottishaw to agree a basic architecture using VoIP simulation and Agilent NgnAS product • (Open OSS Test Bed Components discussion.doc)
14.	12/1/2005		1	Catalyst Charter accepted by TMW board	2) action planning;	2) action planning • TMForum board approved the Catalyst Charter
15.	16/1/2005	18/01/2005	1	Strategic OSS for Next Generation Operators conference organised by IQPC Ltd. London	5) specifying learning	5) specifying learning • Presented OpenOSS catalyst plans at Strategic OSS conference attended by VP level decision makers in the industry • (Strategic Management Craig Gallen;1.pdf) 2) action planning • Significant discussion with Vodafone Germany OSS prime Joerg Frankenberger who agreed to champion OpenOSS project into Vodafone. (BT and Vodafone had just reached a co-marketing agreement to cooperate)
16.	02/02/2005	08/02/2005	1	Team Action Week Lisbon	2) action planning;	2) action planning • Week of detailed planning with participants then signed up for catalyst • BT, Agilent Technologies, AutoMagic KB LLC, Budapest University of Technology and Economics, Cognizant, Invocom, Sun Microsystems, University of Southampton • (TAW Notes_PI_CG comments 09-02-04.doc) 5) specifying learning • Presented work done to date and findings to all the participants at Team Action Week • (OpenOSS_TAW_CatalystWalkThrough-0_1.pdf)
17.	16-2-05		1	meeting to align MDA / Open OSS Catalysts	3) action taking;	3) action taking; • Workshop with Andy Smith QinetiQ, Nektarios Georgalas to create MDA models for OpenOSS which would be used in MDA Catalyst
18.	14-3-05		2	Future Research proposal Outline Strawman	1) diagnosing: 2) action planning;	1) diagnosing: • Reviewing options for further research beyond OpenOSS catalyst 2) action planning; • Worked with Jane Hall FOCUS / Tien Van Do BUTE to outline proposal for FW6 Funding • (OpenOSS FW6 Strawman proposal; 14-03-04.doc)

	From	To	Project	Intervention	AR Phase	Details
19.	09/02/2005	9/05/2005	1	Open OSS Catalyst Development	3) action taking;	3) action taking; • Budapest University of Technology and Enterprise o OpenNMS OSS/J Qos PM Portal Development - BUTE • University of Southampton o OpenNMS OSS/J Qos FM interface o Drools Correlation engine o Test and runtime environment • Agilent o NgnAS OSS/J interface o Cognizant o Integration
20.	11/4/2005		2	OpenOSS Planning Meeting BT Keybridge House London - Monday 15th April, Chairperson: Gary Bruce	2) action planning;	2) action planning; • SUN, BT, Agilent, Focus, Invocom, Cognizant • Discussed possible future directions • discussed FW6 Future Research proposal Outline Strawman • Discussed OSS/J membership (Phillipe Lalande) • Discussed Working with FOCUS Open IMS team (Jane Hall)
21.	12/4/2005		1	Presentation to Vodafone at Vodafone Uxbridge	5) specifying learning 2) action planning	5) specifying learning • Presented OpenOSS program to Vodafone technical primes as arranged by Joerg Frankenberger and Kevin Scarr (Principal Systems Developer in the UK) 2) action planning • Vodafone made a significant contribution to University's costs in participating in catalyst
22.	9/5/2005	13/05/2005	1	Pre-show staging - Southampton University	3) action taking; 2) action planning;	3) action taking; • Last minute integration and testing • Marketing / Presentation run-through
23.	16/05/2005	20/05/2005	1 2	Telemanagement World Nice 2005 Open OSS Catalyst	1) diagnosing: 2) action planning; 3) action taking; 4) evaluating 5) specifying learning	3) action taking; • Catalyst Program presentations and demonstrations - 5 / days over 3 days 4) evaluating • Telemanagement World Nice 2005 wrap up session - 19/5/05 5) specifying learning • Presentations / Booth talks / BOF Session; • all participants talking about experiences 2) action planning; 2 • Telemanagement World Nice 2005 -next steps session 18/5/05 • With Directors from BT, Agilent, TMForum, Vodafone, COLT, C&W
24.	23/6/05		3	Initial discussions with Nortel about Open Source contribution	1) diagnosing: 2) action planning;	1) diagnosing: • Was contacted by Paul Sherrat (Director OSS Platforms - Nortel and discussed possibility of Nortel contributing code to OpenOSS project. 2) action planning; • proposed alternative lines of action Paul Sherratt • (Letter-Paul Sherratt 23-6-05.doc) • (NortelMSP_opensource23-6-05.doc)

	From	To	Project	Intervention	AR Phase	Details
25.	6/07/2005	9/07/2005	1 2	OpenNMS meeting - NC	5) specifying learning 2) action planning;	2) action planning; • planned with OpenNMS for support for summer student project on integrating OpenNMS Alarm List into OSS/J 5) specifying learning • Presented results of OpenOSS catalyst to OpenNMS team
26.	10/07/2005	13/07/2005	1	Presented paper at FOSSTEC 2005, Orlando BT publish internal research on open source	5) specifying learning	5) specifying learning • presented at Free/Open Source Software, Technologies and Content (FOSSTEC 2005) • 'The OpenOSS Project - An open Source operations Support Lab for the Telecommunications Industry' (*Gallen, C. R. et al. 2005c*) • BT publish 'The potential for open source software in telecommunications operational support systems' in BT Technology Journal (*Bruce, G. L. et al. 2005a*)
27.	July 2005	Sep 2005	2	Summer Student work - BT	1) diagnosing 3) action taking; 4) evaluating 5) specifying learning	1) diagnosing; 3) action taking; 4) evaluating 5) specifying learning • During summer of 2005 Southampton a student who participated in Catalyst did investigation into Open Source CRM system for BT and jointly published a paper • SugarCRM investigation (*Bruce, G. L. et al. 2006*)
28.	July 2005	Sep 2005	2	Summer Student work - Agilent / University	1) diagnosing 3) action taking; 4) evaluating 5) specifying learning	1) diagnosing 3) action taking; 4) evaluating 5) specifying learning • During summer of 2005 Agilent sponsored a student to do some open source work with the OSS/J inventory interface (Simon Utting) • University of Southampton sponsored a student (Nick Dance) to continue work on integration OSS/J with Alarm List feature of OpenNMS

29.	25/7/2005		2 3	Monday 25th July
				Location: BT Centre (London)
					2) action planning;	2) action planning;
						• Meeting to determine future direction of Open Source in TMForum
						  ○ TMForum Andrew Chalmers
						  ○ BT Gary Bruce, Paul Robson,
						  ○ Agilent – Doug Carson,
						  ○ Invocom – Simon Pearson, Paolo Imbalzano, ;
						  ○ HP – Nick Parkyn,
						  ○ Vodafone – Mark Adey, Joerg Frankenberger, Andreas Buschmann,
						  ○ Colt – George Whitehouse,
						  ○ UoS – Craig Gallen,
						  ○ Nortel – Jean-Marie Calmel, Paul Sherratt
						• (Notes from OpenOSS Meeting - 25th July 2005-GB.doc)

30.	10/08/2005		4	Initial meeting with Arqiva (Bob Potter) to scope Digital TV work	1) diagnosing:	1) diagnosing: • Meeting to determine if OpenOSS project could offer any help to Arqiva for digital TV work • Initial proposal for work
31.	16/08/2005		4	Initial meeting with Arqiva and Invocom to scope Digital TV work	1) diagnosing: 2) action planning;	1) diagnosing: • Discusses further problems of Digital TV management 2) action planning; • Planned with Invocom for engagement with Arqiva

	From	To	Project	Intervention	AR Phase	Details
32.	10/10/2005	12/10/2005	2	JBoss World Barcelona http://www.jboss.com/events/jbossworld/recap	5) specifying learning	5) specifying learning • Craig Gallen - presentation with Doug Carson Agilent - Reporting on work done over summer with Students. • 'Using JBoss and the Open Source Approach in the OpenOSS Project' (*Gallen, C. R. et al. 2005a*)
33.	18/10/2005		3	Nortel present at OpenOSS weekly call. Offer possibility of Data Management layer	2) action planning;	2) action planning; • Worked with Nortel (Graham Calvert, Miguel Planas, Jean-Marie Calmel and Nigel Davis) to suggest positioning of Nortel Technology in Open Source project. • Graham Calvert presents at weekly OpenOSS call 18/10/2005 • (OOSSData Mgmt.ppt)
34.	31/10/2005	4/11/2005	4	OpenNMS Developers conference Dev-Jam 2005 Pittsboro, NC http://www.opennms.org/index.php/Dev-Jam_2005	2) action planning; 3) action taking; 4) evaluating 5) specifying learning	2) action planning; • Determining how to use the Alarm List functionality with OpenNMS going forwards. 3) action taking; • Working with OpenNMS developer community to understand the forwards plans to use SPRING in the architecture 4) evaluating • Learning about OpenNMS team through local participation; who they are & what they do 5) specifying learning • Presenting the OpenOSS project - explaining the role of OpenNMS
35.	07/11/2005	11/11/2005	3	Telemanagement World Dallas 2005 BOF - NGOSSoil	3) action taking; 4) evaluating	3) action taking; • Launched Nortel NGOSSoil proposal call for participation at BOF in TM World with Graham Calvert / Dave Reberry (CH2M) • Introduced OpenNMS Senior Architect - Matt Brozowski to key players in TMForum 4) evaluating • Evaluating response to Nortel proposal. Planning for Team Action Week
36.	05/12/2005	8/12/2005	2	DMTF WBEM Developers Conference Santa Clara CA http://www.dmtf.org/newsroom/newsletter/2005/09/page2	1) diagnosing:	1) diagnosing: • Comparative analysis of the DMTF developers conference. Learning about WBEM and Cisco's involvement in open source & modelling • Met Steve Jerman Cisco author of Model Wizard (http://sourceforge.net/projects/modelwizard/) • Met Chris Hobbs, Nortel Author of A practical approach to WBEM/CIM management (*Hobbs, C. 2004*) • Met Martin Kirk Program Director Open Group Open Pegasus
37.	27/1/2006		4	Meeting held at Sci-sys Chippenham to discuss Arqiva's proposed Proof of Concept trial.	2) action planning;	2) action planning; • first meeting to plan full proof of concept trial with Invocom, Sci-sys and Arqiva • (Minutes of 2nd Meeting held at Sci.doc)

	From	To	Project	Intervention	AR Phase	Details
38.	29/1/2006	3/2/2006	3	Telemanagement Forum Team Action Week Portugal 2006	2) action planning;	Team Action Week OpenOSS sessions Day 1 • Session 1 Governance Requirements – see Gary Bruce slides • Session 2 – Marketing (see Graham Calvert slides) • TAW_OpenOSS_Meeting1_30-1-05v0.0.1.doc Day 2 • Review of Living Lab Proposal (Craig Gallen Slides) • Review of OIL project Colin Ashford • Methodology for OpenOSS Use case development • Potential scenarios session • TAW_OpenOSS_Meeting2_31-1-06v0.0.1.doc Day 3 • Nortel OIL overview (Graham Calvert Slides) • TAW_OpenOSS_Meeting3_1-2-06v0.0.1.doc Day 4 • Nortel NGOSSoil Deep Dive (Jean Marie Calmel Slides) • TAW_OpenOSS_Meeting4_1-2-06v0.0.1.doc Day 5 • General conversation on way forward • TAW_OpenOSS_Meeting5_2-2-06v0.0.1.doc
39.	3/4/2006	7/4/2006	3 4	NOMS 2006 Panel Discussion Experience Paper	5) specifying learning	5) specifying learning • Panel Discussion Participation 5th April 2006 15:40 - 17:20 Direction of Open Source in OSS Implementation Chair: Alpna Doshi/ Satyam (Systems Integrator) Panellists: Paul Grantham/ Covad; Jenny Hwang/ AT&T; David Reberry/ CH2M; Mike Kelly/ TMF; John Witgreffe/ BT • Paper presentation Gallen, C.R.; Reeve, J.S., "Using Open Source to Realise an NGOSS Proof of Concept", 2006 IEEE/IFIP Network Operations and Management Symposium - Application Sessions - NOMS 2006, April 3-7, 2006 (*Gallen, C. R. et al. 2006*)
40.	12/4/2006		3	Visit to Nortel Ottawa	4) evaluating	4) evaluating • Review Nortel's open OSS Source Strategy with Nortel OSS team • Nortel OA&M product strategy: Miguel Planas, Bill Bourne; Pierre Tremblay; Chris Hobbs, Colin Ashford
41.	24/4/2006	28/4/2006	4	OpenNMS Professional Services - Arqiva	3) action taking;	3) action taking • Taurus Balog provides 1 week professional services in support of OpenNMS in Arqiva PoC
42.	27/4/2006		3	NGOSSoil Decision to close project	4) evaluating	4) evaluating • (Email from Graham Calvert April 27, 2006) • SPLC presentation fails to arouse interest from Service providers. • Nortel officially take their offer off the table to contribute code to OpenOSS in light of lack of support from Service providers

	From	To	Project	Intervention	AR Phase	Details
43.	28/4/2006			Digital TV Proof of Concept Review of results	4) evaluating 5) specifying learning	4) evaluating • Digital TV Proof of concept Review of Progress meeting with key stakeholders; Arqiva, Invocom, Scisis, OpenNMS, Sidonis • (ArqivaDSOPOC_22-4-06.ppt)
44.	23/7/2006	28/7/2006		OpenNMS developers conference Dev-Jam 2006 University of Minnesota	2) action planning; 3) action taking; 4) evaluating 5) specifying learning	2) action planning; • Determining plan to move to MAVEN build process 3) action taking; • Integrating OpenNMS SPRING implementation with OSS/J interface 4) evaluating • Learning about OpenNMS team through local participation; who they are & what they do 5) specifying learning • Presenting the work done with Sidonis at Arqiva in Digital TV management
45.	28/9/06	4/10/06		Meeting with OpenNMS Pittsborough NC	3) action taking;	3) action taking; • Working with Dave Hustace of OpenNMS to integrate Alarm List with latest build of OpenNMS
46.	16/11/2006			Completed OSS/J interface for OpenNMS	3) action taking;	3) action taking; • completed final integration of OpenNMS build with OSS/J interface • Documentation and code on Sourceforge
47.	30/11/2006	3/12/2006		Handover OSS/J interface to OpenNMS team	3) action taking; 4) evaluating	3) action taking; • Final handover of interface - education for OpenNMS developer team and passing over documentation etc 4) evaluating • Evaluating the overall effectiveness and usefulness of OSS/J interface
48.	4/11/06	7/11/06	2	TMW Dallas	4) evaluating	4) evaluating • Observation of state of TMForum after Open Source interventions
49.	22/01/2007	26/01/2007	2 4	Telemanagement Forum Team Action Week Portugal 2007	4) evaluating 3) action taking;	4) evaluating • Looking at evolution of OSS/J and Prosspero ;Tigerstripe training 3) action taking; • Working with QoS team to verify approach to Qos Management in Arqiva • Working with SID team to verify use of SID in Arqiva
50.	20/5/2007	24/5/2007	2	Telemanagement World Nice 2007	4) evaluating	4) evaluating • Brief qualitative survey of leaders of Catalyst projects at TMW to provide a backdrop to findings from OpenOSS catalyst • (SurveyQuestionsTMW2007.doc)
51.		To Present data	1 2 3 4	Final Write up of EngD Thesis	5) specifying learning	5) specifying learning • evaluation of results and extracting specific learning from Action Research projects

Appendix B
Meeting Invocom's Research Objectives

Appendix B Meeting Invocom's Research Objectives

This Appendix describes how the researcher determined with Invocom that their stakeholder expectations would be best served by investigating NGOSS using Open Source and that an interpretive research methodology such as Action Research potentially provided a research paradigm closely aligned with their business model.

B.1 Eng D Research

The traditional mode of industrially based PhD research did not appear broad enough to holistically address the issues present in the OSS industry but the emerging Engineering Doctorate (EngD) mode of study did appear to provide a suitable vehicle. The EPSRC engineering doctorate best practice guidelines (*EPSRC 2006*) note that the purpose of EngD programs are to provide a mechanism and framework for high quality collaboration between academic groups and a range of companies in order to contribute to the body of knowledge on a particular technical discipline, industrial sector or multidisciplinary theme. In this context, the Research Engineer is expected to develop competencies in the chosen technical discipline and in R&D, Product Development, Marketing, Program Management, Teamwork and leadership skills and the ability to apply these skills and knowledge to new and unusual situations. Not only did the multidisciplinary nature of EngD research appear attractive when considering the OSS problem space but it also permitted the research to be partitioned into separate projects addressing an overall theme. This potentially allowed us to legitimately engage longitudinally in projects involving different partners and still deliver an overall coherent result.

At its heart, the EngD requires an industrial sponsor who can provide access to the problem space. In this case the sponsoring company Invocom (*Invocom 2007*) were a small consultancy business having Network and OSS design practices whose clients included a number of service providers in Europe including Orange, BT and C&W. In addition to the industrial sponsor, we determined that developing a close relationship with a key industry body, the Telemanagement Forum (TMForum) (*TMForum 2007f*) would open doors for collaboration with other industry players.

The following sections will describe the process by which the researcher worked with the sponsoring company to converge from a very open research interest in OSS towards a more tightly defined research theme which would be academically sound, of relevance to the industry and useful to the sponsoring company.

B.2 Meeting Sponsor Stakeholder Expectations

At the start of the Eng D program, it was clear that research funding was a new departure for Invocom and it would be important to ensure the work undertaken clearly aligned with Invocom's business needs. As a small SME, Invocom were not in a position to waste money

on irrelevant projects and it was clear that in addition to the sponsorship, Invocom would have to put in additional resources to ensure that the work was best used by their business.

In order to manage the risks associated with the project, a Program Plan (*Gallen C.R. 2003a*) was developed and agreed with Invocom which documented the process by which common agreement would be reached on the research directions for the project. The document used Chapman and Ward's (*Chapman, C. & Ward, S. 2003*) methodology for initiating a project by identifying the aspirations of the key stakeholders, the technology area for investigation and the key risks which the project would face.

The Program Plan also identified that the research would have both business and technical content requiring application of what Cryer (*Cryer, P. 2002 (pp 78-81)*) terms the 'Traditional' research philosophy associated with engineering disciplines and the 'Interpretivist' research philosophy associated with business disciplines. This implied applying business judgement to the interpretation and analysis of technical results and using a 'weight of evidence' approach to reaching conclusions rather than expecting clear cut answers.

In addition it was realised that an Action Research strategy would best align with the needs of Invocom's consultancy business (*Saunders, M. et al. 2003 (pp 93)*). Action Research is a class of research drawn from business or consultancy research which implies direct involvement with practitioners in order to effect a change – either in industry practice or in the business direction of the sponsoring company. Invocom wanted the research to 'create a splash', both by involving a number of Invocom's present or potential customers and by generating enough interest to raise the profile of Invocom across the industry.

Having agreed the broad objectives of the program, a survey of the state of the art of OSS was undertaken in order to identify areas where specific research could be conducted. At this point it was thought that the broad theme of the research could be stated in the following terms;

> 'To investigate the suitability of a particular set of OSS technologies for managing a particular set of 3G services over their supporting 3G networks and leading to the following deliverables;
>
> a. A methodology for specifying the system
> b. An analysis of the validity of the approach and potential improvements
> c. A business analysis of the findings
> d. Potentially a Reference Implementation'

The Technology Report (*Gallen C.R. 2003b*) provided an initial survey of the business challenges of managing evolving service provider networks and services, the current key network management initiatives in the industry and the potential of open source as a disruptive technology for developing OSS software. The document also positioned a number of research choices for further investigation. Key among these choices were;

- Should the research target the DMTF's Web Based Enterprise Management (WBEM) (*DMTF 2005*)initiative or the TM Forum's New Generation Operations Support Systems (NGOSS) initiative (*TMForum 2004b*)?
- Should the research target investigating OSS through Java (OSS/J) (*OSS/J 2007b*) as an implementation of NGOSS.
- Should the research look at Open Source as a potential market discontinuity.
- Which network technology / service set should the research target?
- Which operational pain points should the research target?

Having identified the broad problem area it was now important to do a market analysis in order to position more closely where Invocom should target the research. After having done a survey of the technology initiatives in the industry, we wanted to understand which of these initiatives were likely to succeed and what the time frames for adoption were.

A market survey was conducted to examine the publicised activities of Invocom's key competitors and also the public positioning of the key OSS software vendors and Service Providers. It was also important to begin to leverage the considerable collective market experience represented by Invocom's consultants. To this end a series of 'Strategy meetings' were organised with the Invocom OSS practice in order to elicit feedback on the findings to date. (Strategy Presentations (*Gallen C.R. 2003c*;*Gallen C.R. 2003d*)).

As a result of this consultation, we concluded firstly that, while WBEM did not yet have the mind share of service providers, NGOSS was causing a market discontinuity and an opportunity for new entrants. Secondly that J2EE had already gained considerable market share and was eclipsing .NET in the service provider OSS space. Thirdly, that it was important to understand OSS/J as a J2EE implementation technology for NGOSS. Several recommendations were made for Invocom to consider as outlined in Table 40

In addition three research opportunities were identified.

- Firstly, the DMTF and FM Forum were embarking on a program to align the WBEM CIM model with the NGOSS SID model. There might be an opportunity to get involved.
- Secondly, Invocom were partnering with a product called EMC SMARTS[56] which was based on the DMTF CIM but which was actively addressing the service provider market. There might be an opportunity to look at how SMARTS fitted into an NGOSS framework.
- Thirdly, the notion of Open Source which was initially investigated in the Technology Report as an enabler for further research into NGOSS looked very interesting.

On further consideration, while the first two research opportunities might be interesting, it was felt that the third option of Open Source could create the greatest market impact and also address the recommendations in Table 40. By participating in an Open Source implementation of NGOSS, Invocom would develop expertise in NGOSS, could create a showcase NGOSS implementation and could begin to understand how this framework addressed the problems of their service providers customers.

[56] EMC SMARTS http://www.emc.com/products/software/smarts/smarts_family/index.jsp [Accessed November 2007]

Table 40 Strategic Recommendations for Invocom	
Recommendation	Requirement
• Invocom should learn how to deploy NGOSS	Need capabilities in • SID Information modelling • eTOM Process modelling • NGOSS Lifecycle Implementation & Project Management
• Invocom should be able to position an NGOSS reference framework	• Industry table stakes : o Need to have capability to offer a reference framework of COTS products with OSS/J interfaces. • Gain industry mind share o Be able to demonstrate how today's frameworks could migrate to full NGOSS compliance.
• Invocom should do further market analysis to determine:	• What is the key network service set which will drive new management capabilities? • Which Invocom addressable customers are in the market to participate? • Which products / ISV's should we work with to deliver a reference framework?

B.3 Choosing an applicable Research Methodology

As discussed in Section 1.8, the research methodology chosen must allow the researcher to address the question of;

> 'whether the considered and strategic adoption of open source could further the practice of Telecommunications Operations Support'.

Therefore the aim was not to conduct an investigation into the specifics of good OSS design but rather to consider how open source could make a unique contribution as an aspect of the design process.

Clearly this question could not be approached through the design of a simple laboratory experiment. It required a research methodology which would address the question from multiple perspectives ranging from purely technical to business considerations and facilitate meaningful engagement with the variety of players (researchers and practitioners, individual and corporate) in both the Open Source and OSS communities.

In a 2004 analysis (*Glass, R. L., Ramesh, V., & Vessey, I. 2004*) noted that the most common academic subdivisions of research in computing disciplines are computer science (CS), software engineering, (SE), and information systems (IS). They presented a taxonomy of 19 categories of computing research methods Table 41 which could be identified as having been used in the literature.

Table 41 computing research methods (after Glass, R. L. et al. 2004)		
• action research • conceptual analysis • conceptual analysis/mathematical • concept implementation (proof of concept) • case study • data analysis • descriptive/exploratory survey	• ethnography • field experiment • field study • grounded theory • hermeneutics • instrument development	• laboratory experiment - human subjects • laboratory experiment - software • literature review/analysis • mathematical proof • protocol analysis • simulation.

However (*Holz, H. J., Applin, A., Haberman, B., Joyce, D., Purchase, H., & Reed, C. 2006*) noted that the taxonomy was not useful for computer science as almost 90% of the computer science papers they reviewed were categorized as either conceptual analysis/mathematical

(73.4%) or conceptual analysis/non-mathematical (15.1%). Software engineering had a similarly limited breadth of methodology. Only Information Systems (IS) research covered the breadth of techniques listed in that all but one of the categories were populated and no more than 2% of the entries were placed in any single category. They explained this by noting that IS has a strong tradition of researching research methods and as a result, IS has begun to approach some sort of consensus on terminology and meaning for IS research methodology.

One learned society for the promotion of IS research (The UK Academy for Information Systems) defines the Information Systems domain of study as involving;

> 'the study of theories and practices related to the social and technological phenomena, which determine the development, use and effects of information systems in organisations and society' (*UKAIS 2007*)

and scopes the domain of IS study to include;

- Management of information systems and services
- Integration of information systems with organisational strategy and development
- Design, Development, implementation and maintenance of information systems
- Organisational, Social and Economic effects of Information and Communication Technologies (ICT)

This suggested that any study into the use of Open Source in OSS development could usefully draw on methodologies previously used in the discipline of Information Systems (IS) research and of these, Action Research, Concept Implementation (proof of concept) and Case Study as delineated by (*Holz, H. J. et al. 2006*) in Table 42 seemed most appropriate;

Table 42 Definitions of research methodologies
(after Holz, H. J. et al. 2006)

Action Research	"a specific research approach in which the researcher generates new social knowledge about a social system, while at the same time attempting to change it." (*Kock, N. F. 2004*)
Case Study:	"Single Case: examines a single organisation, group, or system in detail; involves no variable manipulation, experimental design or controls; is exploratory in nature. Multiple Case Studies: as for single case studies, but carried out in a small number of organisations or context." (*Alavi, M. & Carlson, P. 1992*)
Concept Implementation:	"implementation of a system, as in proof of concept" (*Vessey, I., Ramesh, V., & Glass, R. L. 2001*)

It was necessary to structure this work in such a way that it could be defended as having validity for the OSS community. This issue came sharply into focus when the researcher received comments on a paper submitted to a network management symposium to the effect that it was 'too marketing focused'. Without commenting on the merits of the paper, there does appear to be an issue in that there are fairly rigid expectations as to what is considered to be valid OSS research.

This is not a unique issue to OSS research. (*Glass, R. L. et al. 2004*) point out a very real problem between the computing disciplines in that 'each of the fields has its own set of preferred research approaches and research methods, which do not necessarily command the respect of the other disciplines'. One suspected that this problem was likely to effect any cross-disciplinary research into OSS.

(*Avison, D. et al. 1999*) note that in several IS literature surveys prior to 1997 Action Research is barely mentioned as an explicit topic. However they suggested that many papers found, for instance, in the Communications of the ACM discussing the lessons learned from particular projects (variously described as case studies, systems design, software engineering projects etc.) might loosely be classified as of an "action research type," even though the term "action research" is never used in the article. They posit that the value of such articles might be enhanced if the cycle of Action Research were adhered to and described explicitly in the context of the particular projects being discussed. One could thus conjecture that the above observations may also be true of case studies and proofs of concept reported the field of Network Management/OSS.

Most of the formal research published concerning Network Management/OSS is positivist in nature i.e. it is in the nature of empirical investigations into the applicability of a technology or technique. This is illustrated by a search of the journals most often publishing papers addressing the present scope of concern i.e. the Telemanagement Forum NGOSS initiative. A literature search on NGOSS reveals that the majority of the peer reviewed research contributions to NGOSS appear in the following sources;

- Proceedings of the IFIP/IEEE Symposium on Integrated Management (*IFIP/IEEE 2007*)
- Proceedings of the IEEE/IFIP Network Operations & Management Symposium (*IEEE/IFIP 2007*)
- Journal of network and Systems Management (*NwSysMgt 2007*)
- BT Technology Journal (*BT 2007a*)

Many of the papers in these sources could be classified as reporting on proofs of concept and so this mode of enquiry is reasonably commonplace. However the picture becomes more blurred when one looks for case studies and almost non existent when one looks for Action Research[57].

As stated in Chapter 1, it is worth further noting that there are a whole body of conferences, papers and activities such as those sponsored by the TMForum which are not considered 'academic' but which significantly influence the practical direction of the OSS industry.

The question of how to assign validity to truth claims made in trade association environments falls more into the purview of a business school discipline such as Strategic Management where the characterisation of the posturing and business objectives of competitive organisations is the field of study (see for instance (*Thompson, A. A. & Strickland, A. J. 2003*)).

---

[57] A preliminary search using the ACM Portal showed that of 369 articles cited from the Journal of Network and Systems Management there were no references to 'action research' and only 48 'case studies'. A similar search using IEEE Explore to look at the Proceedings of the Network Operations and Maintenance Symposium showed that of 711 citations there was 1 'action research' study (ours (*Gallen, C. R. et al. 2006*)) and 11 'case studies'. A more general search of all citations on IEEE Explore revealed that only 74 of 1540244 documents referenced 'action research'.

A search of the BT Technology Journal revealed that, of 898 citations, there were 89 case studies and 2 action research studies. (Interestingly the most recent action research referenced detailed a forthcoming action research project being planned by BT to look at Security risk mitigation for information systems (*Page, V. et al. 2007*) perhaps suggesting that action research may be coming more into the purview of telecommunications researchers).

Since many Information Systems researchers are co-located in business schools, it is not surprising that they take time to consider the epistemology or theory of knowledge underlying their research. This was an important consideration in this research as it would be engaging with both academic and trade association environments and so one needed to be able to handle both appropriately. Furthermore, as shall be argued, the introduction of open source potentially bridges the gap between pure research and product development and potentially makes truth claims and their underlying assumptions more transparent.

Appendix C
Impact of Open Source on eTOM Processes

Appendix C Impact of Open Source on eTOM Processes

Figure 26, Figure 27 and Table 43 describe which generic eTOM processes (*TMForum 2007b*) (and where applicable their SEI COTS Process equivalent's (*Brownsword, L. et al. 2000*)) are potentially affected by the introduction of open source.

Figure 26 identifies the process areas in the eTOM which are likely to be concerned with COTS introduction and which are therefore likely to be modified by open source introduction. The relationship between the processes directly influencing COTS or open source introduction is brought out more clearly in Figure 27.

This analysis only considers which processes affect the generic introduction of open source to a service provider. Other processes will change depending on the application functionality.

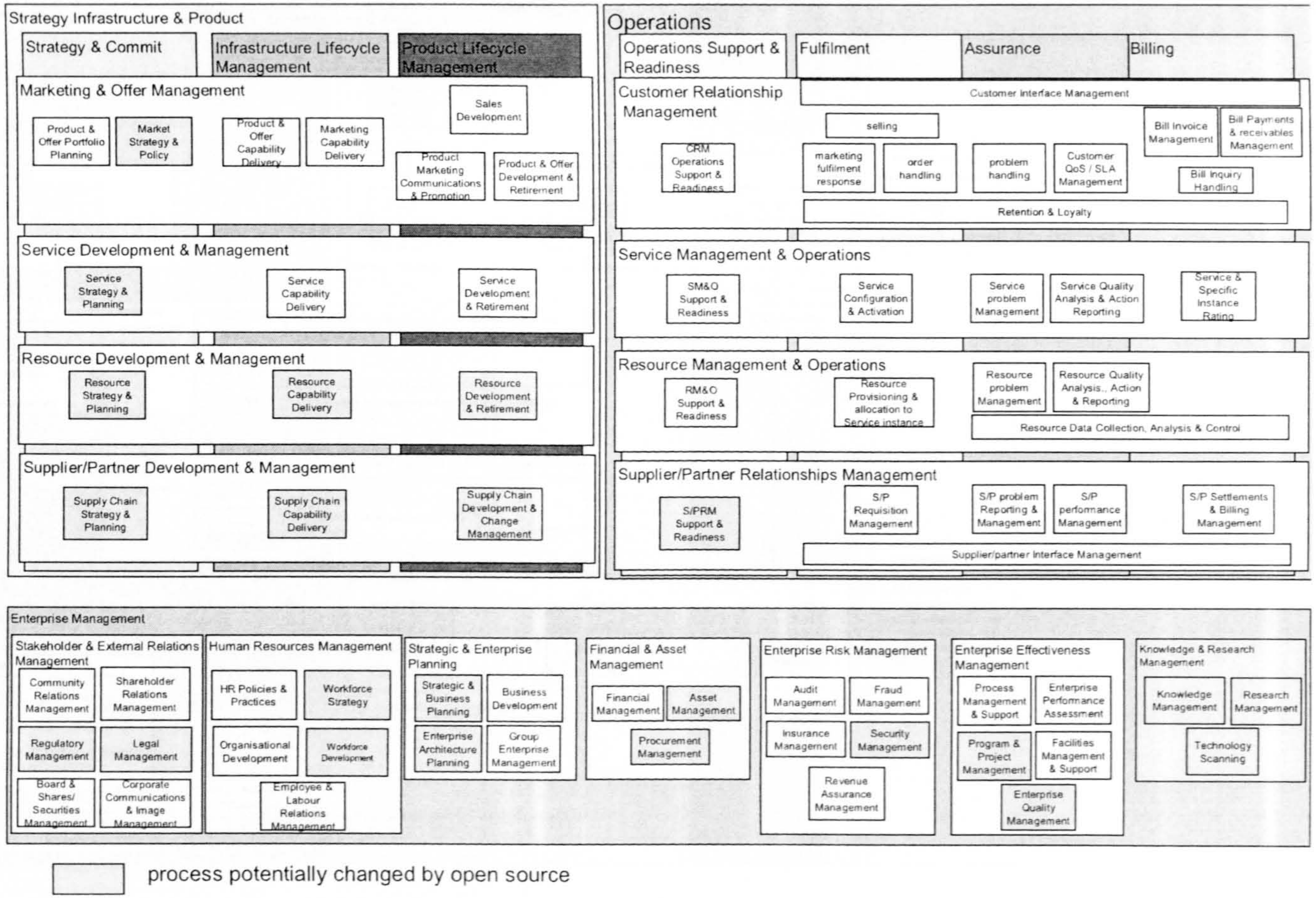


Figure 26 eTOM Processes directly influencing COTS or open source procurement

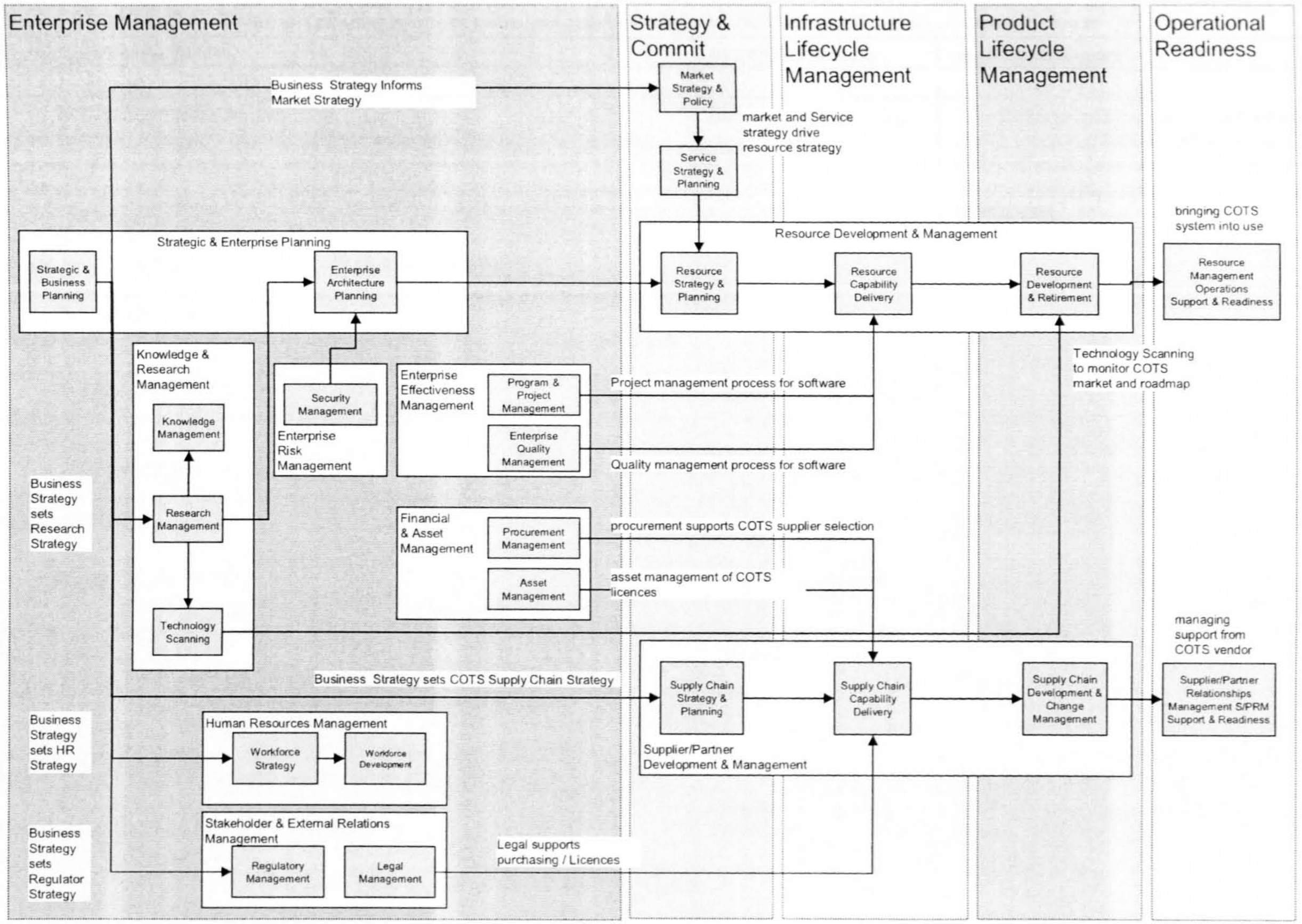


Figure 27 eTOM Processes relationships directly influencing COTS or open source procurement

Table 43 eTOM process areas potentially affected by the introduction of Open Source		
eTOM Level 2 process short description (*TMForum 2007b*)	**SEI COTS process**	**open source impact**
STRATEGY		
Strategic Business Planning Process Identifier: 1.3.1.1 • These processes encompass all of the functions required to provide strategic business direction to the enterprise	Business Activity: COTS business case	If the business believes that the wider business environment is embracing open source - as suggested by writers such as (*Castells, M. 2000*) and (*Benkler, Y. 2006*). The company needs to develop a long term strategy to embrace this new environment. This strategy needs explicitly stated and worked out across the organisation
Market Strategy & Policy Process Identifier:1.2.1.1 • Enable the development of a strategic view of an enterprise's existing and desired market-place.		A market strategy which embraces open source will find ways to increase partners' use of 'open source toolkits' to bring the service providers network infrastructure closer to end customer requirements.
Service Strategy & Planning Process Identifier:1.2.2.1 • Enable the development of a strategic view and a multi-year business plan for the enterprise's services and service directions		The strategy for service development will embrace open source to address and prototype early adopter requirements in order to cost effectively develop the service before mass market adoption.
Supply Chain Strategy & Planning Process Identifier:1.2.4.1 • Develop the Supply Chain strategies and policies of the enterprise, and the enterprise policies for supplier/partner engagement and interaction.	Business Activity: Vendor relationships	As vendors increasingly use the open source model to develop their products and trial standards, the service providers must find ways to actively engage with this process as part of the supply chain management process.
Workforce Strategy Process Identifier:1.3.7.3 • These processes create the strategies needed to ensure that the correct type, quantity and quality of employees will be available in right locations for future business.	Project-Activity: Cultural transition	An Open Source bias will look to develop a workforce which has an active interest in participating in open source projects. It may even recruit staff because of their external involvement in strategic open source objects
DEVELOPMENT		
Enterprise Architecture Planning Process Identifier:1.3.1.3 • The business strategies and plans provide a key starting point for the development of the Enterprise Architecture. • Define, develop, manage and maintain the enterprise architecture, which is the model describing the enterprise in all its facets covering both IT and network technologies. • Provides the reference source for the enterprise Standard Operating Environment that is the collection of applications, hardware, software and network components standardised for use across the enterprise. • Define/select, develop and deploy the enterprise architecture methodology, • Define and select the toolsets used to capture the enterprise architecture (e.g. UML tools). Define, develop and maintain the repositories used to capture architecture details, and manage and administer the capture processes.	Project-Activity: CBS strategy Project-Activity: CBS trade-offs	As open source becomes increasingly viable, the role of enterprise architecture planning becomes one of clearly articulating the data and information standards across the organisation and encouraging the selected open source projects to support these standards. By contributing toolkits, documentation and key developers the Architecture role can more directly influence the future product direction of the industry and the adoption of a technology by their own organisation. The role of the Architecture team becomes much more tangible because of their influence exerted on their vendor's open source product development.
Research Management Process Identifier:1.3.4.2 • Plan and prioritize research and development activities within the enterprise. • Identify and evaluate the business value of R&D investment. • These processes cooperate with Technology Acquisition for corporate-level decisions on "make vs. buy" concerning internal or external Research & Development investment.	Engineering Activity: Evaluation	The Research management process becomes one of contributing use cases and limited resources to open source projects in order to learn how to use emerging technology
Knowledge Management Process Identifier:1.3.4.1 • Knowledge Management processes are responsible for managing the tacit and explicit knowledge contained with the enterprise. • Patent management, Intellectual Property Rights management,		The role of patents and IP becomes less confrontational under the open source model. The organisation needs to understand much more clearly what are their key assets and what can they afford to share within the open source development community. The organisation also needs to protect itself against any potential IP or licence infringements.

Table 43 eTOM process areas potentially affected by the introduction of Open Source		
Technology Scanning Process Identifier:1.3.4.3 • Technology Scanning processes perform scanning and initial assessment of emerging technology from external sources for the enterprise. This includes identification and evaluation of potential technologies, and sources of technologies, and comparison with the "in-house" research capabilities. It also includes determining the business value of potential technology acquisitions, and provides an initial assessment of any financial or other commitments to actually acquire specific technologies where this is appropriate. • These processes cooperates with Research Management processes for corporate-level decisions on "internally develop vs. buy" concerning emerging technology, by providing expert assessment assistance in making the decision. Any sourcing of the proposed technology is delivered through the Supply Chain Capability Delivery processes		The technology scanning process increasingly includes participation in open source projects in order to understand what other parts of the industry are doing and in order to learn how to use emerging technologies. The organisation needs to be able to allow researchers to download and trial software.
Resource Strategy & Planning Process Identifier:1.2.3.1 • Develop resource strategies, policies and plans for the enterprise, based on the long-term business, market, product and service directions of the enterprise. • Understand the capabilities of the existing enterprise infrastructure, capture the requirements on the infrastructure based on market, product and service strategies of the enterprise', manage the capabilities of the suppliers and partners to develop and deliver new resource capabilities, and define the way that new or enhanced infrastructure may be deployed.	Engineering Activity: COTS Evaluation Business Activity: COTS cost estimation Business Activity: COTS business case	Increasingly working with equipment vendors to understand the roadmap for products and use open source to help understand how to manage new features emerging in the network.
DELIVERY		
Program & Project Management Process Identifier:1.3.3.3 • Define and manage the program and project management methodologies, skills and tools within the enterprise. • Projects are actually managed within other areas of the enterprise but follow the guidelines defined by this process	Project-Activity: CBS risk management	Project management needs to understand better how to manage the uncertainty associated with roadmaps in open source projects. Need to adopt ideas of 'risk efficiency' in order to be faster to market with projects having less certain outcomes.
Workforce Development Process Identifier:1.3.7.4 • Workforce Development processes focus on development of employees to meet the needs of the business.		Training, mentoring and active encouraging of staff into open source activities.
Supply Chain Capability Delivery Process Identifier:1.2.4.2 • Manage the evaluation of new supplier/partners, to determine those with the products that best meet the enterprise's needs • establish commercial arrangements with selected suppliers/partners. These processes manage the activities associated with developing tender documents, running tender processes, gaining enterprise agreement to tender decisions, and negotiating commercial arrangements as a result of the tender decisions.	Business Activity: Vendor relationships	Developing the supply chain for taking open source developments and turning them into supported products. How much development and packaging and support in house How much outsourced to suppliers
Resource Capability Delivery Process Identifier:1.2.3.2 • Use the capability definition or requirements to deploy new and/or enhanced technologies and associated resources. • The objectives of these processes is to ensure that network, application and computing resources are deployed, according to the plans set by Resource Development. • Handover the Resource Capability to Operations	Engineering Activity: Construction	Clear process for managing risk when moving from high risk open source projects towards low risk production systems. Internal secure open source repositories for production systems code and distribution management

Table 43 eTOM process areas potentially affected by the introduction of Open Source		
PRODUCTION		
RM&O Support & Readiness Process Identifier:1.1.3.1 • Manage resource infrastructure to ensure that appropriate application, computing and network resources are available and ready to support the Fulfilment, Assurance and Billing • Supporting the operational introduction of new and/or modified resource infrastructure and conducting operations readiness testing and acceptance • Managing planned outages; • Performing pro-active maintenance and repair activities;	Engineering Activity: Configuration management Operational Readiness Testing	First second and third line support for imported open source software and for hybrid solutions Bug and Patch management
Supplier/Partner Relationships Management Support & Readiness Process Identifier:1.1.4.1 • Ensure that all necessary facilities related to the interaction with suppliers and partners are ready and functioning. Moreover, these processes are responsible for the resolution of problems related to these facilities.	Engineering Activity: Deployment and maintenance	The granularity of external interaction goes up with open source as the organisation needs the ability to interact with relatively small project teams as well as large suppliers.
SUSTAINING		
Resource Development & Retirement Process Identifier:1.2.3.3 • Develop new, or enhance existing technologies and associated resource types, so that new Products are available to be sold to customers. These processes also retire or remove technology and associated resource types, which are no longer required by the enterprise.	Engineering Activity: Deployment and maintenance	Long term support arrangements. How to support open source technologies when the original developers have left the project. Which projects must we support from internal resources What do we do about technologies which rely on underlying projects.
Supply Chain Development & Change Management Process Identifier:1.2.4.3 • Supply Chain Development & Change Management processes support the development of a supply chain, to support service catalogue extension or modification. • These processes particularly apply when the commercial arrangements between the two parties exist over an extended period, and the commercial agreement allows for the on-going specification of deliverables within the period of the commercial agreement.	Business Activity: Vendor relationships	Long term commercial arrangements for the support and development of open source software. Note that the security of supply depends on keeping the supply chain whole - not on extracting the best possible price (free).
SUPPORTING PROCESSES		
Regulatory Management Process Identifier:1.3.6.4 • Ensure that the enterprise complies with all existing government regulations		Need to be clear about any anti-trust issues Need to be clear that open source policy is not a means to skew the market
Legal Management Process Identifier:1.3.6.5 • Ensure that the enterprise complies with all relevant legal requirements, • Supporting the enterprise by providing legal advice related to business decisions	Business Activity: :License negotiation	Understanding open source licences and IP issues
Asset Management Process Identifier:1.3.5.2 • Manage all financial and policy aspects of the physical assets of the enterprise		Open source artefacts do not show up as assets. Need different financial models to understand the value of licence free software
Procurement Management Process Identifier:1.3.5.3 • Define corporate procurement and logistics policies and rules, for all purchases, • Manage and control the acceptance of goods from suppliers		Need better models for valuing the return on an investment in terms of the future development options offered by open source.
Security Management Process Identifier:1.3.2.2 • Assess threats to the enterprise and provide capabilities and controls to minimize the threats identified. • Actual security monitoring, control and management procedures and facilities are embedded in the operational infrastructure and processes defined and deployed within the SIP and Operations process areas.		Need clear testing and security guidelines for software and systems procured under the open source model.

Appendix D - Open OSS Stakeholder Expectations

Appendix D Open OSS Stakeholder Expectations

This appendix provides a detailed analysis of the motivations of the stakeholders in the OpenOSS catalyst.

D.1 ISV Stakeholder Expectations

Three ISV's directly supported the OpenOSS project; Agilent, Sun and Automagic.

Middleware vendor Sun Microsystems supported the project through their membership of the TMForum Board and provided help with using OSS/J. Sun were generally favourable towards open source and although they had not at that time open sourced Solaris and Java, they were well on the way towards publicly adopting an open source strategy. Their primary motivation for founding OSS/J as a J2EE implementation of NGOSS was to encourage the adoption of enterprise middleware into service providers' OSS Stacks; the argument being that product development in the OSS industry needed to leverage the much greater R&D investment going into the global e-commerce industry.

Sun's perspective was that open source provided a good vehicle to prototype OSS solutions using emerging standards which could subsequently be realised using commercial implementations. Sun encouraged the project to use open source implementations of standards developed through the JCP (*JCP 2004a*) including;

- JSR 94 Java Rule Engine API (*JCP 2004c*)
- JSR 168 Java Portlet specification (*JCP 2003b*)
- OSS/J - various OSS interface standards developed through the JCP process

From a marketing perspective, their view was that OSS/J was already a proven standard and that any academic involvement should not be positioned as research which called into question the technical design of OSS/J. Rather the objective should be to investigate if open source could help increase the adoption of OSS/J by the industry.

Interestingly only one COTS OSS ISV (Agilent) volunteered to be involved and, perhaps understandably, their involvement was predicated on not having a direct competitor also involved in the project - although they would have welcomed complementary products (such as an inventory system). Agilent wanted to use the project for three purposes.

Firstly, Agilent's Marketing prime wanted to use the project to understand the extent to which open source could potentially commoditise functionality within their products. This was based on their reading of Clayton Christensen's work which had suggested that open source was a disruptive technology (*Christensen, C. M. 2003*), (*Christensen, C. M. 2004*). Although initially open source projects could not compete in functionality against COTS OSS products, if they continued to innovate, eventually their functionality would become 'good enough' and customers would switch to the open source equivalent. A COTS vendor needed to be aware of this danger and plan when to 'move up the value chain'.

Secondly, Agilent's Architecture prime wanted to investigate the use of open source to build a test harness which simulated complementary products. In this context, OSS/J was seen as a

standard interface which would be used to link Agilent's products to other vendors' products in future real projects. Agilent also wanted to use the project to begin some collaborative investigations with students at the University of Southampton into other uses of OSS/J. This ultimately resulted in post Catalyst work using the OSS/J Inventory interface which was presented at JBoss World (*Gallen, C. R. et al. 2005a*)

Finally, although not the primary objective, the work was also justified to their management as a vehicle for engaging with customers and providing a platform for showing off their NgnAS (*Agilent 2002*) product at the Telemanagement World Trade show.

Automagic KB LLB, were a small company providing training and consultancy in the use of NGOSS. They were in the process of creating a methodology and UML toolset for using NGOSS to design real systems and wanted opportunities to develop their own understanding and promote their work. Automagic offered to model the OpenOSS project and subsequently used it as an example in their NGOSS training material.

Although the OpenNMS (*OpenNMS 2007c*), OTRS (*OTRS 2007*) and Drools (*JBoss 2007b*) projects had commercial open source ISV's offering support options, they were not directly involved in the project. It was assumed that their commercial models would not have supported a project funded out of 'cost of sales' and they would have wanted financing to get directly involved. In practice, however, when the projects were contacted they were very helpful and interested. (The story of the increasing involvement of the researcher with OpenNMS is dealt with separately in Chapter 6).

It was decided to treat the Open Source components used in the Catalyst as Closed Source components in that the Catalyst did not intend to develop enhanced functionality which could be contributed back to the projects. This represented GCBA development (Table 17) and provided the opportunity to test Tactical Question 1 in Section 3.5.2.1 and Tactical Question 3 in Section 3.5.2.2; looking at the issues of Estimation, Adaptation and Support (but not the issue of Security) listed in Table 19.

D.2 Integrator Stakeholder Expectations

In some senses, the two Integrators involved in the OpenOSS catalyst, Invocom and Cognizant were in competition. However their business models were sufficiently different to allow them to co-exist in this project.

Invocom are a niche player based in the UK who provide advanced consultancy in network design and in the deployment of OSS systems. They seek high value business based upon fostering trusted customer relationships as project architects or project managers rather than competing directly on cost at the implementation level. Invocom were the long term sponsors of this researchers EngD program and wanted to use the project to demonstrate thought leadership in the industry. Invocom lent an experienced project manager in order to keep the project on track.

Cognizant are a much larger Indian based integrator whose basic business model is to leverage their access to low-cost off-shore development resources in order to win business in American and European service providers. They have had a long term strategy to resource catalyst projects in order to build credibility, develop customer contacts and learn about emerging technologies[58].

Cognizant wanted an opportunity to showcase their Indian Open Source practice in the project integration role. Cognizant's involvement was also useful as it potentially gave some insight into the Emerging markets discussed in Section 3.4.2.

D.3 Non Market Institutions (Universities and TMForum) Stakeholder Expectations

Two Universities were involved in the project, the University of Southampton (UoS) and the Budapest University of Technology and Business (BUTE). The Universities wanted to use the project to create industry based educational opportunities for students. Both institutions wanted to increase the relevance of their research to industry and ultimately if possible attract research funding.

As discussed in Section 3.6.2, the TMForum has a complex role as the principal non market institution in the OSS industry. The TMForum has the ultimate objective of demonstrating value to its members through innovative projects and realisable standards.

The TMForum is not entirely neutral in that it is an agent of its membership who pay for its services. Thus to sustain its activities, the TMForum must keep its members happy in order to maintain its revenue stream from membership fees. Membership privileges include access to published TMForum material and the right to participate in the TMForum working groups. The open source angle posed a problem for the TMForum since it normally restricted access to NGOSS related projects to TMForum members whereas the true open source model requires unrestricted access.

Although the Service Providers were demonstrably interested in investigating open source, the majority of the revenue in the TMForum comes from the COTS ISV members who made it plain that they did not want to see the TMForum sponsoring any open source projects which might compete with some aspect of their business. The OpenOSS catalyst had to respect this unresolved tension and this is reflected in the construction of the OpenOSS Catalyst Charter (*TMForum 2004h*) which mandated looking primarily at developing open source glue technology and not at applications.

---

[58] Strategy confirmed in an interview recorded at Telemanagement World May 20-24 2007 with Sajith Sankar Cognizant Leader of 'NGN OSS Blueprint' Catalyst, http://www.tmforum.org/RecentlyCompletedProjects/NGNOSSBlueprintCatalyst/32494/article.html [Accessed Sept 2007]

D.4 Service Provider Stakeholder Expectations

The service providers showed considerable interest in investigating open source as demonstrated by the fact that seven service providers allowed their names to be used as the sponsors of the OpenOSS project (BT, Vodafone, Covad, QinetiQ, C&W, Colt and NTT).

BT was the major contributor although Vodafone, who for reasons of the timing of an internal re-organisation could not get directly involved, made a financial gift to the University of Southampton to cover their expenses.

In considering the power structures in the client-system (see Section 2.1.4), it is clear that BT carried the most influence and that the other partners became interested primarily because BT were involved. It is very useful therefore to first consider BT's overall technical strategy as the context in which they engaged with this work. BT is particularly useful to study since it actively publishes the thinking behind its technical strategy in a way which is less common in the other UK operators. BT's research interests, now collected under the BT Exact organisation, derive from BT's history as a UK PTT. The other UK based operators (including Vodafone) have tended to rely on their suppliers for R&D and do not have equivalent organisations (This corresponds with the outsourcing of R&D discussed in Section 3.4.2). This diagnosis section will concentrate on understanding BT's motivations and contrast these, were relevant, with the motivations of the other service providers.

One of the mechanisms BT uses to signal its strategic intent to its stakeholders (suppliers, customers, shareholders, partners and the internal organisation) is through its research at BT Exact and its publications in the BT Technology Journal. One needs to consider these publications with caution as there can be a disconnect between what the research division thinks and how the company acts (See Section 3.6.1). However BT Exact has in recent years been targeted more closely on problems of immediate strategic interest to the company and Matt Bross BT Group CTO and three other senior directors endorsed the strategy described in papers published in the BT Technology journal in 2005 (*Reeve, M. H. et al. 2005*). This provides a useful window into BT which affords a particular but detailed technical view of the industry against which the published views of other service providers can be triangulated.

(*Reeve, M. H. et al. 2005*) describe BT as having inherited 'Stove Pipe' architecture; a legacy of multiple services, each associated with its own network and support systems. This is a problem described as common to many other operators by (*Lee, J. & Ben Natan, R. 2002 pp 74*). BT's strategy for radical cost reduction is to close its legacy networks and fold the existing services on to a new multi-service network called 21CN. The plan addresses the following concerns.

D.4.1 BT's Network and Service Strategy

British Telecom's flagship technical program for developing their 21st Century Network (21CN) (*BT 2007c*) is designed to significantly simplify their complex multi-technology legacy

network (Figure 28) towards a considerably simpler topology (Figure 29) based upon an MPLS core.

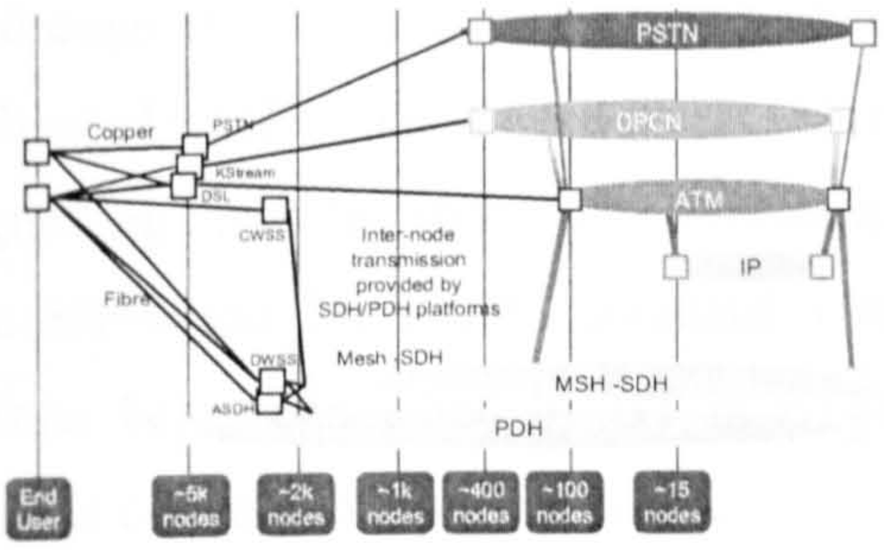


Figure 28 BT's Legacy Network layering

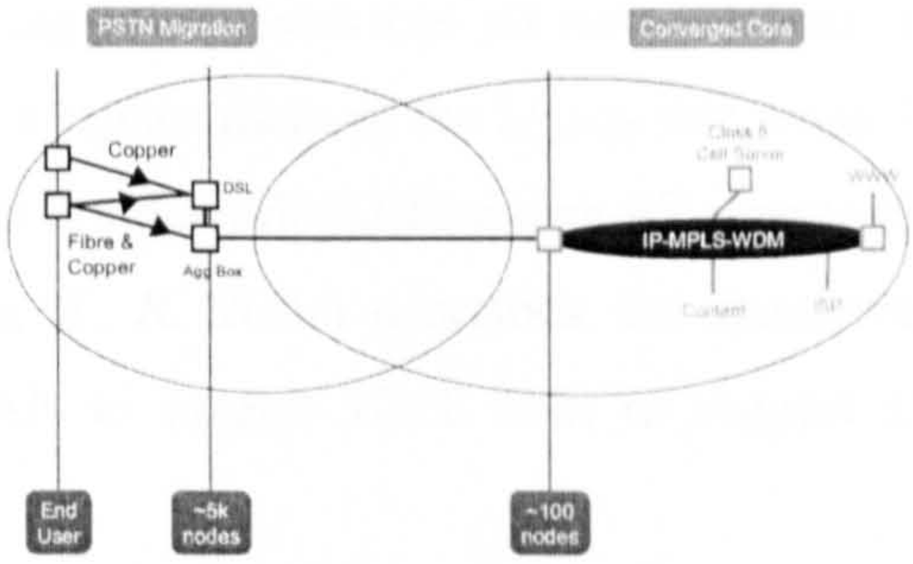


Figure 29 BT's proposed 21CN network

(*Crane, P. 2005*) describes how BT envision the future UK network will be partitioned into 4 types of node; Core, Metro, Multi-Service Access and Customer Nodes (Figure 30).

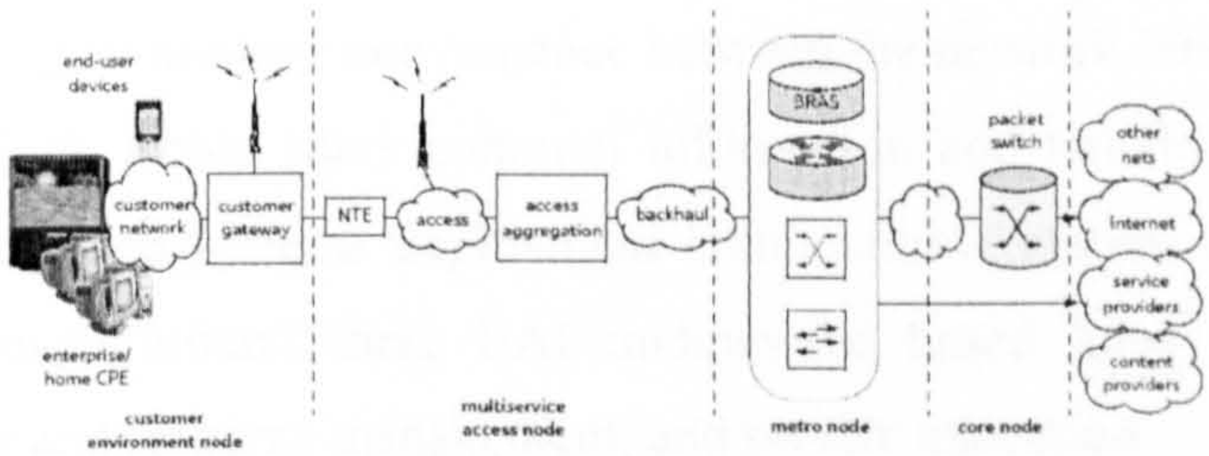


Fig 3 The physical network architecture.

Figure 30 BT Physical NW Infrastructure
(after Crane, P 2005)

In summary, the converged network will carry new and legacy services (such as frame relay, ATM and PSTN) over a core MPLS infrastructure. Ten Core Nodes will terminate a national MPLS core network interconnected by an SDH/optical transport. Each Metro Node will be connected to at least 2 core nodes for resiliency.

Metro Nodes will aggregate the access services in metropolitan areas and adapt them to the core MPLS platform. Metro Nodes will also provide the interconnect points into other provider networks. Metro Nodes will provide connection-oriented aggregation and switching functions and connectionless routing and will also host most of the network's intelligence in the form of multimedia IP service platforms providing call-control signalling and VoIP media transcoding.

Multi-service Access Nodes will simply act as remote aggregators for the Metro Nodes, terminating local cable and fibre on Network Termination Equipment (NTE). Large customers will interconnect using fibre. Single customers will interconnect with copper/DSL.

In this model, Customer Nodes represent all the network devices on the customer side of the NTE. A key aspect of the Customer Node is the Customer Gateway which acts as a service control and management end-point and a service enabler for the customer network. The specifics of the customer gateway will depend very much on the service offered. For instance, (*Galbraith, R. I. 2002*) describes the functions which might be required in a customer gateway which integrates DVB-T with broadband delivered IP-TV. It is important to note that the

Customer Node may only be partially under the control of the network provider and creating an environment where third parties can develop complex Customer Gateways which interact correctly with the service provider network will be important.

Although the network will be capable of carrying legacy services all new services will be delivered by IP at the service level. This requires a replacement of the legacy telephony IN and signalling functions with a new Service Execution Framework (SEF) which BT plan to base on the SIP-based 3GPP IMS standard. (*Dickerson, K. R. 2004*) describes the standardisation efforts being conducted by BT in ETSI TISPAN to extend 3GPP IMS to support a fixed mobile converged (FMC) network.

It is useful to note that Vodafone had recently become a partner with BT for fixed mobile convergence activities. As a 3G mobile operator Vodafone were equally interested in the management of 3GPP IMS infrastructures. Vodafone were also a customer of Agilent's NgnAS product which they were using to manage performance across their mobile network.

BT envisage increasing technology convergence between the network intelligence systems and the OSS systems which would share common information and functions. (*Crane, P. 2005*) states that BT's distributed systems deployment architecture (referred to as the 'integration grid') will be deployed around three EAI middleware based hubs supporting customer management, service and resource management, and service execution.

(*Wardlaw, M. 2005*) describes how the SEF will be used to deliver SIP based services. Currently SIP based VoIP services are implemented using Soft Switches (Figure 31). The SEF will significantly extend these capabilities towards multimedia services (Figure 32). BT envisages a business model where services can be launched by third parties using applications which access the capabilities of BT's network exposed through the SEF. The SEF architecture facilitates this by separating the applications through controlled interfaces from the underlying service-enabling capabilities of the network.

The SEF will be built using a service-oriented architecture (SOA) supported by the same types of middleware as are currently used for OSS integration. Service applications will consume and provide Web Services and thus complex services can be constructed out of other services.

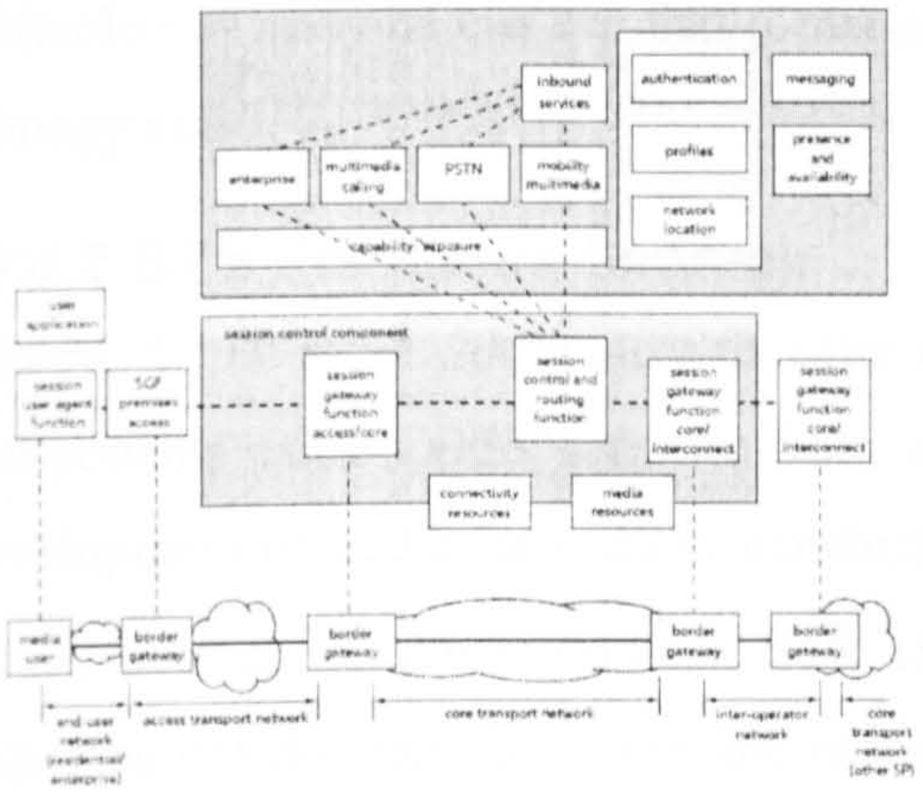


Fig 4 An example of VoIP using SIP

Figure 31 BT Example of SIP VoIP
(after Wardlaw, M. 2005)

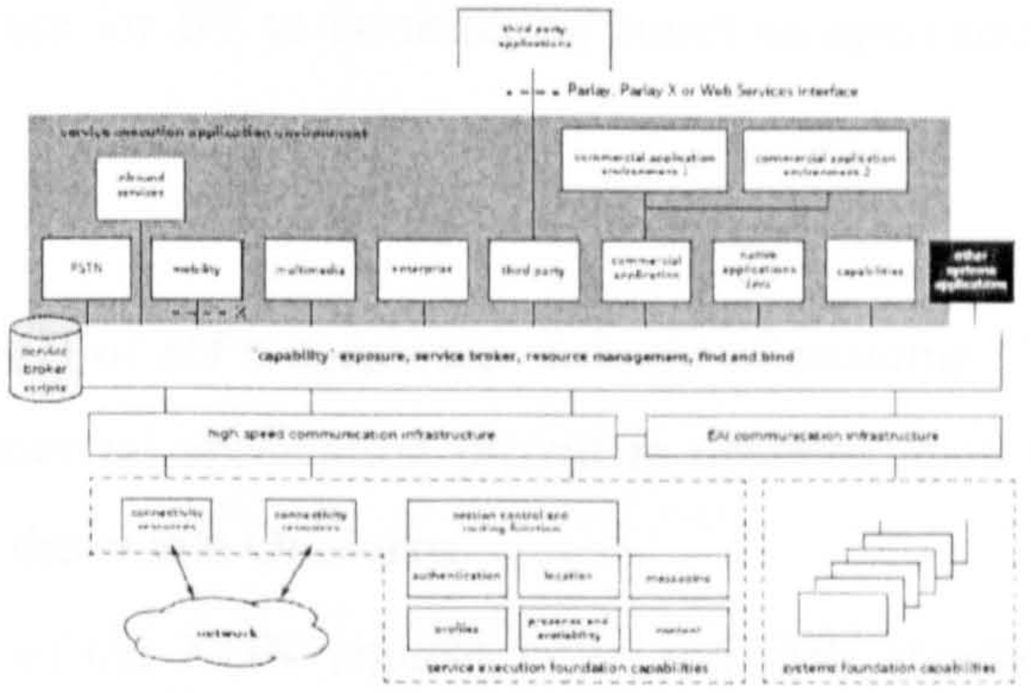


Fig 2 The Service Execution Framework — all applications are supported by reusable capabilities

Figure 32 BT Service Execution Framework
(after Wardlaw, M. 2005)

(*Darling, J. & Sauvage, A. 2005*) describe the envisaged application lifecycle for new telecoms services provided by BT or third parties hosted on BT's Application framework. In the past there were high barriers to entry for innovative telecoms service provision. New services required knowledge of proprietary and complex interfaces, and telecommunications industry-specific development tools. (*Crane, P. 2005*) notes that a key aim for the SEF is to lower the technical barriers to entry for IT developers such that it will be possible to build a community of developers and grow an eco-system of innovative companies currently excluded from the telecommunications market. This will require providing a set of tools which will be intuitive to users of industry standard development tools, such as Microsoft .NET.

To gain some insight as to what this might look like in practice, (*Darling, J. et al. 2005*) note that an early release of this technology has been used to deliver BT Wholesale's Web Call Connect product which has been used by call centre providers. In this product the service is discovered using UDDI and a web service interaction using the Parlay X protocol is used to set up a call between 2 phones. The first phone rings and if answered the second phone rings and the call is completed. A free simulator is available based on the BEA Web Logic J2EE container to assist developers to test applications (*BT 2006*)

BT clearly envisage a change in the way they do business with their partners and the use of a toolkit to allow end users to design services which interact with the complex SEF infrastructure is intended to create new market opportunities. This appears to be a first attempt by a telecoms operator to use toolkits in order to foster innovative service creation by users in what was previously a closed environment. The link between toolkits and open source is quite strong and was discussed in Section 3.6.3 which considered von Hippel's observations on toolkits for innovation (*von Hippel, E. et al. 1995*).

We should note that the decision to offer all new services using IP at the service level will move BT's future business transactions into the business layers above the TCP/IP interface in Fransman's model (See Section 3.4.2). As already noted, this is an area where open source development is already much more common than in telecoms. Add to this the desire to use the SEF to lower the technical barriers to entry for IT developers who will use common e-business

technologies and one can see that it makes sense for BT to be thinking about an open source strategy now.

D.4.2 BT's Standards Strategy

(*Reeve, M. H. et al. 2005*) state that the success of BT's corporate strategy of building OSS components using COTS software from commercial application providers depends upon the development of, and adherence to standards in the whole OSS area.

(*Dickerson, K. R. 2004*) outlines BT's view of the NGN control layer and OSS standards requiring development to reduce the cost of delivering converged networks such as 21CN. In an OSS conference presentation in 2005, Dave Millham of BT (*Milham, D. 2005*) took the view that the remaining technical challenges in the control layer of converged networks were on the way to being solved. However the OSS industry needed to take several steps in advance of the network and mobility control protocols being available in order to address the problems of realising these technologies as services. Although NGOSS had provided a potential framework for realising NGN OSS, the industry needed to go to the next level of trialling how to use these standards to develop realistic solutions against the emerging mobility technologies. This included developing and trialling the capabilities outlined in Table 44.

We can see that BT already seek to strategically influence the OSS market through their involvement in standards bodies like the TMForum. A question to be addressed is, could open source enhance the standardisation process as discussed in Section 3.6.2

Table 44 OSS capabilities required for next generation services	
Service Management	Solutions for scalable management of individually customised services
Common B2B trading model	A common model for managing the complex trading environment of a multi service and network market
Product lifecycle model	A common model for rapidly deploying telecoms products
Common technology management model	A common model for managing multiple communications technologies
Consistent processes	Generic processes which can be applied to any technology
Consistent information models (TMF MTNM, SID, CIM)	Common information models to allow rapid integration of OSS capabilities.

D.4.3 BT's OSS Rationalisation Strategy

BT currently has over 700 major computing applications (*Strang, C. J. 2005*). (*Potter, J. M. M. & Brady, A. R. 2005*) tell us that most of BT's legacy systems have been developed 'in house' to support bespoke specifications but (*Reeve, M. H. et al. 2005*) state the BT's policy is that in the future OSS components will be built using off-the-shelf components from commercial application providers and integrated rapidly using middleware hubs.

(*Reeve, M. H. et al. 2005*) characterise the basic problem which BT faces when using COTS components. Without standards COTS components do not conform to the same functional architecture (i.e. they overlap in what they do) or have the same data model (they do not mean the same thing and use the same description for items such as 'customer'). With standards adoption BT could more easily fit externally developed components into their architecture

without extensive modification and integration activities. (*Crane, P. 2005*) notes that there is real potential for dramatic reduction in integration costs by choosing COTS packages which conform to industry standards for information and process models of which the SID and eTOM are the most relevant examples.

(*Potter, J. M. M. et al. 2005*) outline the process which BT have been undertaking to rationalise investment in their IT infrastructure using an analysis of the business value delivered by systems which combines aspects of the eTOM with Porter's Value Chain (*Porter, M. E. 1998*). This has resulted in BT's BT's Single List of Authorised Systems and Applications (SALSA). One purpose of SALSA is to source an approved list of partner COTS vendors with whom BT negotiate frame contracts.

Although BT does not appear to claim to have used NGOSS explicitly in designing their architecture, they have certainly followed a number of the principles which NGOSS espouses. It is informative to try and map BT's systems rationalisation activities onto the Business, System, Implementation and Deployment aspects of the NGOSS lifecycle (*TMForum 2004a*).

The NGOSS Deployment View appears to map onto BT's distributed systems deployment architecture (the 'integration grid') (*Crane, P. 2005*).

The NGOSS Business View appears to roughly map onto BT's Business Component Architecture which (*Crane, P. 2005*) describes as consisting of Atomic Business Components (ABC's) which represent a collection of people, skills, internal processes, and system capabilities, Each ABC contracts its services to other ABCs, and presents all its services via XML-type electronic service-oriented interfaces.

The NGOSS System View appears to map onto BT's System Capability Framework (SCF). BT has a particular opinion about the functions which each component will perform in their network (which may be a subset of vendor's view of their product's capabilities). This is characterised in BT's 'Matrix architecture' which positions the strategic systems onto a standard functional representation of BT's business which has been derived from the eTOM. The Matrix architecture is further refined in BT's System Capability Framework (SCF) which abstracts reusable and configurable capabilities out of the system functionality (*Potter, J. M. M. et al. 2005*)

(*Strang, C. J. 2005*) describes the SCF as being composed of Reusable Engineering Capabilities which are defined as 'well-known callable services relating to the systems implementation of a high level business operation, capable of being deployed and reused across a broad range of products and business channels to facilitate the rapid deployment of integrated systems solutions'. Capabilities have the qualities listed in Table 45.

BT clearly have an interest in promoting these capabilities into the standards forums and this definition of a System Capability maps very closely to the concept of an NGOSS Contract (*TMForum 2004i*). (*Strang, C. J. 2005*) notes that BT's transformation program plans to

develop over fifty Capabilities in the coming years. Perhaps unsurprisingly, BT has submitted 50 candidate NGOSS Contracts to the TMForum which the Service Provider Leadership Council have asked the newly formed NGOSS Contract Management Program lead by to consider (*TMForum 2006b*)

The NGOSS Implementation view is less visible but (*Strang, C. J. 2005*) provides some insights as to how the System Capability Framework (which was characterised above as the NGOSS Systems View) maps onto implementations built around approved COTS vendors. BT names the major COTS vendors in Table 46 as having a role in their implementation;

The Cramer inventory system will hold a common network model used by network engineering, service fulfilment, and service assurance. Cramer have been working closely with BT in the TMForum to define the MTOSI specification for passing inventory information (*TMForum 2005f*).

The link between the order management system and network intelligence will be via a 'meta-directory' - a set of set of LDAP directories holding the customer/user profiles which are used to set up network policy (somewhat equivalent to the Directory Enabled Network strategy discussed in Section 1.3.4). The intelligence layer will cache this profile and the BEA integration platform will replicate this information into the order management system, Seibel.

Table 45 BT's Reusable Engineering Capabilities
(Strang, C. J. 2005)

• Be mappable to/from the Systems Capability Framework. • Relate to a well understood business operation. • Relate to a reusable, stateless service called either synchronously, asynchronously or both. • Be designed with reuse as a primary requirement. • Generally be business-wide and not product/service specific. • Have a well-defined (XML) API with both hard and soft parameters. • Have well-defined and well-documented error conditions.	• Have well-defined business and system dependencies. • Encapsulate the complexity of the COTS package(s) used to implement it. • Be capable of external publication, even though it may not be made public. • Have a non-functional as well as functional specification, covering availability, performance, support, and testing. • Be designed to accommodate future change without having an impact on existing users. • Be tested rigorously and generically so as to enable black boxing by future users.

Table 46 Major COTS Vendors in the BT's OSS Implementation
(Strang, C. J. 2005)

• CRM order management	Siebel	• Integration hub	BEA J2EE using MQ light, Web Services and EAI.
• CRM fulfilment and outsourcing	Clarify	• Cross-domain performance management:	Quallaby Technology
• Billing	Geneva	• Service Assurance	Still under consideration.
• inventory management	Cramer		

D.4.4 BT's NGOSS Modelling Strategy

(*Strang, C. J. 2005*) reports that BT engineers will use a unified information model for the architecture modelled in a single UML repository and accessed using a common UML case tool. In BT's case, they have adopted Borland's UML modelling tool-kits. (*Azmoodeh, M. et*

al.) describe this as a massive initiative in trying to use an off-the shelf UML tool in a design process which provides a unified approach to requirements capture, system and component design, and centralised model repository management. However (*Azmoodeh, M. et al.*) report that a significant problem with this approach lies in trying to accurately model current COTS products. The lack of precision in the resulting models means that the resulting specifications become open to interpretation.

BT's research direction is to move the industry towards increasingly rigorous NGOSS Contract specifications which can be accurately modelled in UML. Ultimately the aim is to use Model Driven Architecture techniques to generate configurations (policy and workflows) which are instantiated directly into NGOSS components (*Georgalas, N. & Bagley, C. 2005*), (*Yang, K., Ou, S., Azmoodeh, M., & Georgalas, N. 2005*).

One criticism levelled at NGOSS is that it is paper based and too complex for the average OSS Architect to realistically use in specifying a system. Therefore a significant push within the TMForum is to provide tooling and worked examples to help with industry uptake of the standard.

UML offers the potential to model an NGOSS system as a set of components offering contract interfaces which expose data in terms of the SID model which is already created in UML. Specific work flows derived from the eTOM could in theory be modelled as actions on these interfaces using UML Activity Diagrams. Unfortunately several problems still need to be overcome. Firstly, while in theory UML models can be transferred between tools using the XDI standard, in practice transfer between current tools maintains the model but often loses graphical information. This currently forces companies to adopt a UML tool from one vendor to ensure interoperability and also raises questions as to which tool the TMForum should use to maintain the standard (currently IBM's Rational suit - which is quite expensive). Secondly, the tool vendors supporting the eTOM (Telelogic/Popkin [59]) support business process modelling standards and do not create UML.

Thus while UML potentially provides a better tool than paper specifications for NGOSS design, the best practice for using UML is still being explored. The TMForum are working with the OMG to potentially extend the UML meta-model in order to allow Contacts to be better described (*TMForum 2003*). However at present potential users of NGOSS have two options. They could wait until the UML and NGOSS Contact specifications are sufficiently mature to allow modelling by NGOSS Contracts. Alternatively they can take the view (as BT are doing) that UML tools are sufficiently mature and available for use now with some 'work-arounds'.

In order to move the UML modelling and MDA agendas forwards, BT sponsored an MDA Catalyst in parallel with the OpenOSS catalyst. The objective of the MDA catalyst was to

[59] http://www.telelogic.com/ [Accessed Sept 2007]

explore the state of the art in MDA tool interoperability and to demonstrate modelling of an OSS solution using UML and NGOSS artefacts. The aim was to build market awareness through examples and training to show how NGOSS and UML could be used today as part of a practical design process. As part of the MDA Catalyst, Automagic KB undertook to try and model the OpenOSS catalyst using a UML tool loaded with NGOSS SID objects and eTOM workflows.

D.4.5 BT's Open Source Research objectives

During the period leading up to the OpenOSS planning workshop (2/12/2004), BT conducted an investigation into the potential benefits of Open Source for BT which also drew upon the work done by the University of Southampton Students integrating OTRS and OpenNMS during the summer of 2004. The key conclusions of this investigation were shared at the planning workshop and eventually published by BT after the Catalyst (*Bruce, G. L. et al. 2005a*).

The focus of BT's investigation was on understanding the risks associated with deploying open source solutions and gaining an appreciation of the maturity of the open source market place. BT concluded that while there were mature and industry strength infrastructure components (e.g. operating systems, application servers and security tools) open source OSS applications were relatively immature. BT had already chosen a COTS component strategy and was not interested in pursuing an open source application development path.

While BT would eventually like to see the emergence of open source OSS application, their primary focus was on understanding how open source could reduce the cost of integrating COTS products if open source implementations of TMForum interfaces became available. In the longer term, BT recognised that there was potential in creating solutions using an open source software approach combined with OSS open interface standards such as OSS/J which could compete in innovative niche products or SME situations where the COTS OSS equivalent would be prohibitively expensive.

If mature open source OSS components were available, they projected a potential saving of at least 8% by applying open source software over the total life cost of the OSS, where the entire OSS solution used mature open source software with a commercial support arrangement. This was based upon modelling licence and support costs in comparison with COTS products. This signified the potential value of being able to make a risk efficient choice of open source software for niche applications (see Section 3.3).

It is important to note that the focus of BT's interest was on potential cost savings through licence cost reduction and reducing the OSS integration tax imposed by OSS vendors. Their study treated open source components as equivalent to COTS. Their study did not consider any potential advantages of the open source development methodology as a means to facilitate better innovation or requirements management with their vendors.

D.4.6 Other Service Provider Stakeholder Expectations

NTT were already active users of open source in Japan and simply wanted to see the subject addressed within the TMForum. Unlike BT and NTT the other service providers did not have large R&D departments and so there requirements were more project orientated.

C&W was interested in open source as a potential cost reduction vehicle - particularly in their off shore properties.

Vodafone already had a strong commitment to OSS/J (*Buschmann, A. et al. 2007*) but wanted to investigate how collaborative research involving academia in the TMForum could de-risk their architectural decisions. They wanted to investigate creating a vehicle to research how to use emerging standards like OSS/J before requesting them in commercial projects.

Colt wanted to understand the potential of open source to support B2B integration with network operators across Europe. COVAD had already addressed this problem in America using OSS/J to create a trouble ticket interface and wanted to contribute the interface as open source and find other developers within the OSS/J community (COVAD subsequently published the project on Java.net (*COVAD 2007*)).

The desire of the smaller service providers to find efficient ways to develop multiple B2B interfaces to the major suppliers was evidence of the increasing network effects in the telecoms industry as discussed in Section 3.4.1.

Appendix E
OpenNMS Case Study

Appendix E OpenNMS Case Study

This appendix provides a case study and functionality description of the OpenNMS project.

E.1 OpenNMS History

The following history of OpenNMS has been compiled through interviews with Tarus Balog, the current OpenNMS Maintainer and by reviewing early OpenNMS information pre-dating Tarus and still available on the web (*Giles, S. 2000*), (*O'Donnell, S. 2000a*), (*O'Donnell, S. 2000b*), (*O'Donnell, S. 2002*). Table 47 lists the key milestones in OpenNMS's history.

Table 47 Key OpenNMS Milestones	
Jan 1999	Initial OpenNMS 'Bluebird' project begins - architecture designed by Steve Giles and implemented by Brian Weaver
Mar 1999	Built prototype application using IDE
Apr 1999	Finished prototype, began final design work
July 1999	Team quit jobs and moved to small office in Cary, NC and start OcuLAN
Nov 1999	Took EUI alpha code on a road show to several very large network providers in Wash DC, Denver, Florida and Virginia. Received feedback and integrated changes.
Jan 2000	Tested distributed poller discovery on a very large outsourcer, found problems, made corrections
Mar 2000	Announced new project, turned on web site, released source code
Apr 2000	Release first version of Swing GUI. Hired new manager Shane O'Donnell responsible to coordinate project and for communications with the community of 'over 2000'. Management team: • Shane O'Donnell: Open Source Project Manager, 2 years consulting and training, 10 years, network management architect • Brian Weaver: Development Manager, 7 years network management development experience. • Steve Giles: Architect, 11 years network management development experience. • Luke Rindfuss: Coordinator/Marketing, 2 years sales and marketing experience in network management software. Contributors • 10 full-time team members (Including management). • Approximately 50 community members who are active contributors in design, architecture, development, documentation, or testing.
June 2000	jSNMP released SNMPv1 and SNMPv2 libraries for Java (later renamed joeSNMP to avoid confusion with another product http://sourceforge.net/projects/joesnmp/)
Aug 2000	Project gets institutional funding
Nov 2000	Beta test begins
Dec 2000	First release Version 0.4
May 2001	OpenNMS 0.96 released
Sept 2001	Tarus Balog started working for OcuLAN as Support Manager for OpenNMS.
May 2002	OpenNMS 1.0 released OcuLAN decide to leave open source market Tarus Balog becomes maintainer of OpenNMS and founds Sortova Consulting Group to provide support for OpenNMS
June 2003	Tarus Balog joins Blast Internet Services who purchase assets of Sortova [60].
March 2004	David Hustace joins Blast as director of professional services (OpenNMS)
Sept 2004	September 2004, Blast spins OpenNMS off into its own company, The OpenNMS Group, Inc. (David Hustace, President, Tarus Balog CEO)
June 2004	Matt Brozowski joins OpenNMS as CTO and Architect
Oct 2005	Dev-Jam 2005 First OpenNMS Developers Conference
July 2006	Dev-Jam 2006 Second OpenNMS Developers Conference
July 2007	Dev-Jam 2007 Third OpenNMS Developers Conference

Brian Weaver[61] and Steve Giles were the founders of Onion Peel software which created integration plug-in's for HP OpenView (*HP 2007b*). When they sold their company during the

[60] http://www.wral.com/business/local_tech_wire/news/story/1154754/ Blast Internet Services Acquires OpenNMS from Sortova for Cash;

[61] Brian Weaver's profile http://www.linkedin.com/pub/1/79B/221 [accessed 8-9-07]

dot.com bubble to Netplex[62] they wanted to spend their earnings on starting a new network management venture. However they realised that most successful network management vendors had extensive marketing machines with which they could not compete and so they decided to create a niche open source product as a guerrilla marketing tactic. The project was initiated around 1999 and originally named 'bluebird'[63].

Having created an early prototype, the team founded OcuLAN and sought institutional funding allowing them to initially resource the project with 10 staff. In addition the project reported 50 external contributors and 2000 members of the community (*O'Donnell, S. 2000a*). In September 2001 Taurus Balog, who had a background as a consultant on HP OpenView and Micromuse Netcool (*IBM 2007b*) was recruited as a Customer Support Manager for the project.

An early marketing presentation suggested that the business case for OpenNMS was based on accessing some part of the $3Bn services and support market in network management (*O'Donnell, S. 2000b*). However after 2 years of operation, OpenNMS was clearly not generating the returns OcuLAN needed and in May 2002 OcuLAN decided to move out of open source and concentrate on creating network management 'appliances' which leveraged OpenNMS code to provide a management solution in a pre-built 1U device each selling for around $10K. (OcuLAN ceased trading in 2004[64])

OcuLAN held the copyright of OpenNMS and so could close source and fork the code which they wanted in their appliances. However the original OpenNMS project was GPL licensed and therefore could not be taken out of circulation. Realising an opportunity, Tarus asked OcuLAN if they would give him the OpenNMS brand if he took over the obligation to support their OpenNMS customers. OcuLAN agreed and Tarus set up Sortova Consulting in May 2002, working from his home to serve the then four support customers. After the first year this had grown to twenty customers. Tarus briefly partnered with a local ISP, Blast Internet Services before forming OpenNMS Group Inc with David Hustace.

As of July 2007, OpenNMS group has about 100 customers in the following countries; US, France, Germany, Italy, Switzerland, Denmark, Israel, Singapore, Australia, Japan and Canada but the majority are in the USA. OpenNMS is developing a channel arrangement for partners outside the USA who want to have access to the OpenNMS trade mark and offer local 1st and 2nd line support with 3rd line support provided by the OpenNMS group. OpenNMS has partners in the Netherlands, Portugal, Italy and the Far East. Tarus' view is that these

---

[62] http://www.highbeam.com/doc/1G1-19815316.html Netplex Acquires Onion Peel Solutions. PR Newswire Date: October 3, 1997 [accessed July 2007]

[63] http://old.lwn.net/2000/0309/a/opennms.html Announcement: OpenNMS.org & Project Bluebird Wed, 8 Mar 2000 [accessed July 2007]

[64] Management vendor Oculan closes doors By Network World Staff, NetworkWorld.com, 05/25/04 http://www.networkworld.com/news/2004/0525oculan.html [accessed July 2007]

arrangements allow him to attract some revenue from markets which he could not otherwise address and is happy that the majority of the revenues there don't come his way.

OpenNMS group have not revealed their absolute revenues but have grown to profitably employ 4 staff and support the overheads of a small company office in North Carolina and the marketing expenses of attending key Open Source and network management expos across America and Europe. Currently the revenue split among services is; 40% basic support, 10% premium support, 20% customer funded development and 30% on site consulting for customers. Partnership revenue falls within the support revenues. So far there is negligible income from running training or conferences.

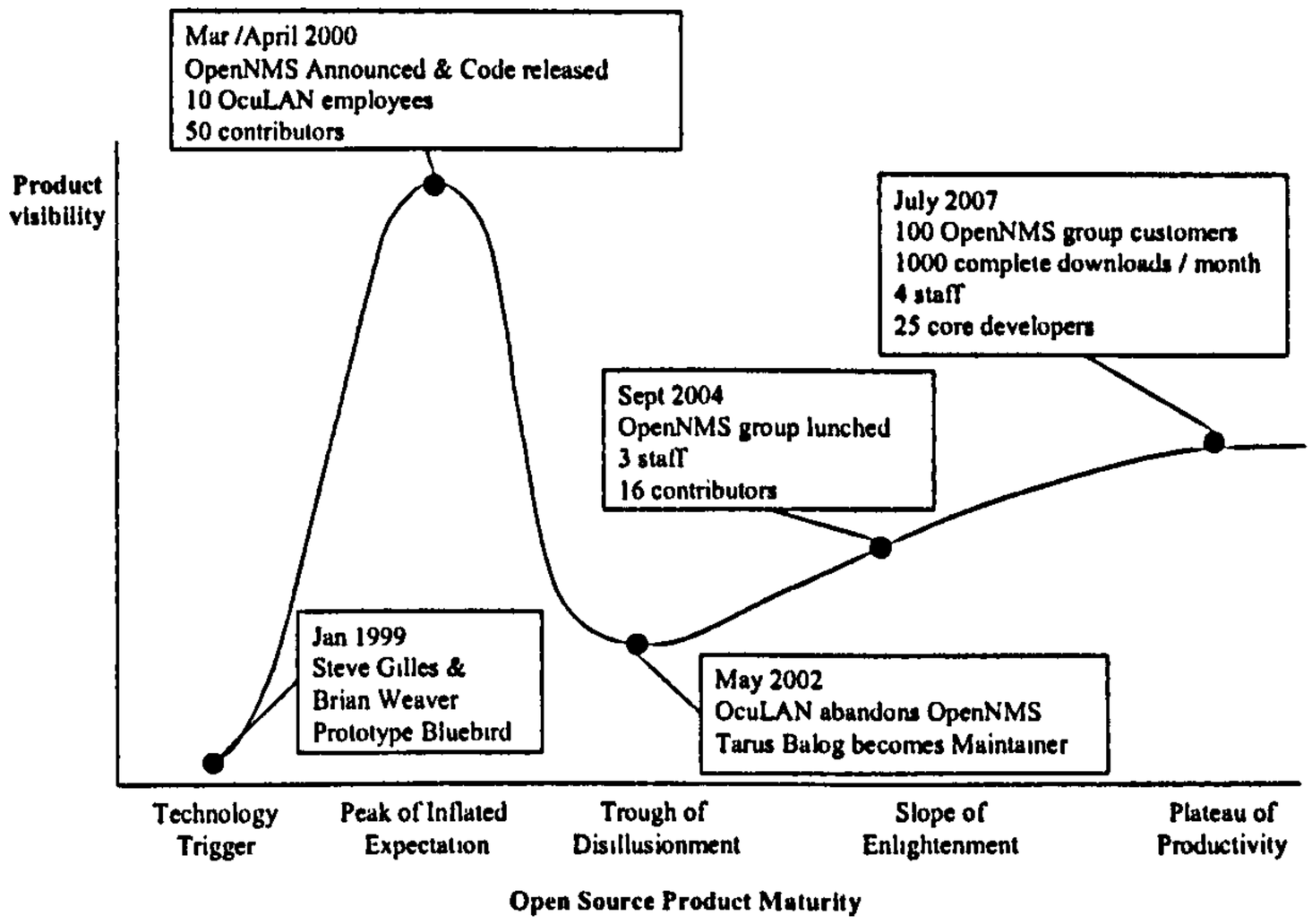


Figure 33 OpenNMS Maturity Curve
(after Bruce, G. L. et al. 2005a)

OpenNMS's history can be summarised by adapting Gartner's open source maturity curve (as used by (*Bruce, G. L. et al. 2005a*)) in Figure 33 where the vertical axis indicates the level of visibility, or hype, in the industry and the horizontal axis indicates maturity as a combination of software stability, range of functionality and overall acceptance by the industry. As can be seen, OpenNMS has reached a point of sustainable maturity and is experiencing increasing adoption as more potential users become aware of its existence.

7.6 OpenNMS as a company within an Open Source Cluster

It is possible that one of the reasons why OpenNMS initially emerged is it was first located in the centre of a significant open source industry cluster. (*Porter, M. E. 1990*) first popularised the notion of industry clusters as an explanation for the competitive advantage of nations. Rosenfeild has applied these concepts to smaller regional clusters to which he has given a working definition of 'A geographically bounded concentration of interdependent businesses with active channels for business transactions, dialogue, and communications, and that collectively shares common opportunities and threats' (*Rosenfeld,*

S. A. 1997). Clusters benefit from specialised infrastructure, labour markets and services which favour the development of particular industries.

OpenNMS originated in North Carolina near the environs of Research Triangle Park (RTP)[65], a major centre of innovation in North America. RTP appears to contain an industry cluster of open source development. It is the home of the major commercial Linux distribution, RedHat[66] and a number of other open source initiatives. Local universities have actively been researching the wider implications of open source. The University of North Carolina sponsors an Open Source Research Team[67] which also runs ibiblio.org[68] and Duke University law school sponsors the Centre for the Study of the Public Domain[69]. At its inception OpenNMS was closely linked with North Carolina State University[70] and are still actively connected with this innovative community[71]. Thus OpenNMS has been able to benefit from local investors, users and developers who were already very familiar with open source concepts.

E.2 OpenNMS Users

It is impossible to know how many OpenNMS users there are but there are in excess of 1000 complete downloads of OpenNMS RPM's a month from Sourceforge. 1500 people subscribe to the discuss list and Tarus assumes that these are all active users. Tarus finds that most people who approach him as a potential customer have already been using or evaluating OpenNMS for some time. Table 48 lists some of OpenNMS's more prominent users, illustrating that OpenNMS has attracted interest from users throughout Fransmans Telecoms layer model. As can be seen, OpenNMS has attracted interest across the telecommunications industry although the likes of Dell and IBM have just been evaluating the technology and recommending further investigation by their developer communities.

---

[65] Research Triangle Park http://www.rtp.org/

[66] RedHat Inc. Worldwide Headquarters 1801 Varsity Drive Raleigh, North Carolina http://www.redhat.com/about/contact/ww/americas/raleigh.html

[67] Open Soruce Research Team (OSRT) http://www.ibiblio.org/osrt/

[68] Ibiblio.org a conservancy of freely available information, including software, music, literature, art, history, science, politics, and cultural studies. http://www.ibiblio.org/about.html#who

[69] The Center for the Study of the Public Domain, Duke University Law School Durham, NC http://www.law.duke.edu/cspd/index

[70] In May 2000 ;Dennis Kekas was appointed Director of Networking Technology in the College of Engineering at North Carolina State University. He was also on the Advisory Board of OpenNMS/Blue Bird Networks http://www.engr.ncsu.edu/news/awards/awards.arc.00.summer.html [accessed 8-9-2007]

[71] For instance, in 2005 the researcher was invited to attend with Dave Hustace from OpenNMS a session on 'OSS: Is Open Closing Out Innovation or Opening Doors for Many' at the UNC Symposium for Intellectual Property, Creativity, and the Innovation Process on 1st -2nd November 2005 http://portal.northcarolina.edu/ipcip/agenda.htm. [accessed 8-9-07]

Table 48 Example OpenNMS Users Within The Telecoms Industry Layer Model				
Telecoms Industry Layer Model (Fransman, M. 2001)		**OpenNMS Application**		
Layer	**Activity**	**OpenNMS User**	**Business**	**Details**
VI	***Customers / Consuming***	Papa Johns Pizza http://www papajohns.com/	Catering	Monitoring network infrastructure
		Minnesota Children's Hospital http://www.childrensmn.org/	Public Hospital and University	Monitoring network infrastructure
		Oregon State University http://oregonstate.edu/	Monitoring network infrastructure	Sites:100 Services:2043
		Permanente Medical Group www.permanente.net/	Large provider of health plans in Northern California	(*Balog, T. 2007b*) Distributed Pollers: 350
V	***Applications Layer, including contents packaging*** (e.g. Web design, on-line information services, broadcasting services, e-commerce etc)	Myspace www.myspace.com	Community Video Content Hosting	Monitoring network infrastructure
		Ocado www.ocado.com	On-line retailer in the UK.	(*Sartin, J. 2005b*) Monitoring Web sales application JMX application monitoring
		FreshDirect http://www.freshdirect.com	On-line retailer in New York	(*Reilly, R. 2007*) Integrated with Splunk www.splunk.com
		Fox TV (Australia) http://www.foxtel.com.au/	Broadcaster	Monitoring DVB-T Transport Stream Analysers
		BBC Monitoring www.monitor.bbc.co.uk	Broadcaster	Monitoring network infrastructure
		Arqiva	Broadcaster	Proof of concept for DVB-T Network
IV	***Navigation & Middleware Layer*** (e.g. browsers, portals, search engines, directory assistance, security, electronic payment, etc)	FastSearch http://www.fastsearch.com/	Internet search services	Monitoring search engines
III	***Connectivity Layer*** (e.g. Internet access, Web hosting)	New Edge Networks http://www.newedgenetworks.com/	IP VPN network provider	(*Brunelli, M. 2005*) (*Balog, T. 2007b*) Providing customer service stats Nodes: 12278 Interfaces: 67738 Services: 569188 (300,000 data points every five minutes)
		Rackspace http://www.rackspace.com	Managed Hosting	(*Winslow, M. 2004*) Nodes:200
TCP/IP INTERFACE				
II	***Network Layer*** (e.g. optical fibre network, mobile network, DSL local network, radio access network, Ethernet, frame relay, ISDN, ATM, etc)	Swisscom Eurospot www.swisscom-eurospot.com/	monitoring European Wifi network	(*Balog, T. 2007b*) Sites: 2373 Nodes: 4760 3 Interfaces: 50001 Services:61601
		BT www.bt com	Satellite data services	Monitoring Satellite services at Jodrell Bank
		Opera Telecom http://www.operatelecom.com/	SMS voting services for broadcasters	Monitoring network infrastructure
		TruePhone http://www.truphone.com/	SIP based Wifi mobile telephony provider	Monitoring network infrastructure
I	***Equipment & Software Layer*** (e.g. switches, transmission equipment, base stations, routers, servers, CPE, billing software etc)	Dell www.dell.com	Example using OpenNMS to manage Dell servers	Application Note (*Giles, J. 2003*)
		IBM www.ibm.com	Example using OpenNMS to manage JMX	Application Note: (*Li, S. 2002*)

The majority of OpenNMS Group's customers would fall into the 'enterprise customer/consuming' segment. Generally these unsophisticated customers have chosen the product over COTS offerings based on price and purchase only the basic support package. Apart from initial installation they rarely have issues with the system although Tarus finds that

his more troublesome customers have issues because they do not have staff sufficiently trained to understand any complex network management application.

When they started the business, Tarus assumed that all of the interest would come from the enterprise consumer segment however most of the paid-for development has actually come from more sophisticated users in the lower layers of Fransmans model. In the case of some, such as Rackspace and Swisscom and Ocado, the choice was made based upon proven scalability and the fact that they did not have to pat a 'per-interface' licence fee which priced other COTS products out of their market. In the case of New Edge Networks, the choice was based upon the ability to integrate OpenNMS into a customer portal in order to report VPN performance to customers. In this case, OpenNMS has been used to create a premium customer facing service.

All of the more sophisticated customers have staff who have acted as OpenNMS champions and participate in the project in various capacities. Some customers have paid the OpenNMS group to do custom development (e.g. Kaiser Permanente, Swisscom) but in other cases they have dedicated staff who do local customisations and contribute work back to OpenNMS (e.g. Jonathan Sartin in Ocado). Although OpenNMS is a business critical application for these users, they have variously managed the risks of using an open source application by either taking a support contract or having developers on staff who contribute to the project. It is clear that in many of these cases, the perceived risk of open source has been far outweighed by the financial benefits or the fact that they could not have cost effectively developed the same functionality using a COTS product.

E.3 OpenNMS Organisation and Developers

Sourceforge lists 25 developers who have commit rights to OpenNMS, of which there are at least 16 core developers who are recognised as key contributors. These are identified as 'members of the Order of the Green Polo' (OGP), a humorous accolade termed by Tarus Balog[72].) The majority of the volunteer developers are employed by companies who use OpenNMS to manage their infrastructure although some, such as Antonio Russo in Italy, are independent consultants who use OpenNMS as part of a solutions sale to their customers. Interestingly, the majority would not consider themselves to be professional programmers but either systems architects, systems administrators or networking engineers. Their expertise is in running systems or networks and OpenNMS is their tool of choice.

All the employees of OpenNMS group are designated officers in the company as this is a way for a small firm to avoid certain employee obligations under US law. The roles of the 4 employees are listed in Table 49.

[72] http://www.opennms.org/index.php/Order_of_the_Green_Polo

Table 49 Roles of OpenNMS Group employees	
Tarus Balog CEO	• Marketing / Sales - 'ability to talk technically to non technical people' • Attendance at Linux World trade shows etc; • First line customer support on production systems • Bug fixing • Admin / Finance etc
Dave Hustace President & Chief Operating Officer	• Requirements specification for customers of new features • Solution delivery project management • Operationalising support relationships with partners and customers • Programming
Matt Brozowski CTO	• OpenNMS architectural direction • New technology evaluation and introduction • Developer community mentoring and training • Programming
Ben Reed CIO	• Build system and packaging • Systems admin for development systems, bugzilla, and wiki • Programming (Ben just joined company in 2007 - but was with the original OpenNMS project and has maintained a Fink (Apple Mac) distribution of OpenNMS in the intervening period)

E.4 OpenNMS Design

Although the OpenNMS wiki (*OpenNMS 2007c*), installation guide (*OpenNMS 2007b*) and release notes (*OpenNMS 2007d*) provide some information about the design of OpenNMS, the majority of the documentation (*OpenNMS 2007a*) is aimed at helping the OpenNMS user. The researcher found that he had to rely primarily on analysis of the code and on discussions with the OpenNMS team in order to get a useful understanding of the internals of the product. The following description summarises this analysis.

Figure 34 provides an overview of the functional architecture of OpenNMS with an indication of the major enhancements which have been introduced in the last two years. OpenNMS consists of a server component which runs as a Linux daemon service and a Web UI component which runs in a separate Tomcat JSP server (*Apache 2007a*). The system can be distributed across several machines for scalability.

E.4.1 OpenNMS Server

The OpenNMS server hosts a variety of Java daemons which run as separate threads within the JVM. An open source implementation of the Java Management Extensions (JMX) standard (*SUN Microsystems 2006b*) called MX4J (*MX4J 2007*) is used by OpenNMS for internal process management. Each OpenNMS daemon is a JMX bean which is initialised, started, stopped and monitored by the Controller process which bootstraps the OpenNMS server at start-up. The status of the daemons can be interrogated using JMX and viewed using a JMX web client internal to MX4J. The Log4J framework (*Apache 2007d*) is used to log errors or notable lifecycle events separately for each of the daemons. The configurations for the daemons are contained in properties files and XML files stored in the OpenNMS /etc directory. The XML configurations are loaded using the open source Castor XML marshalling framework (*Castor 2007*).

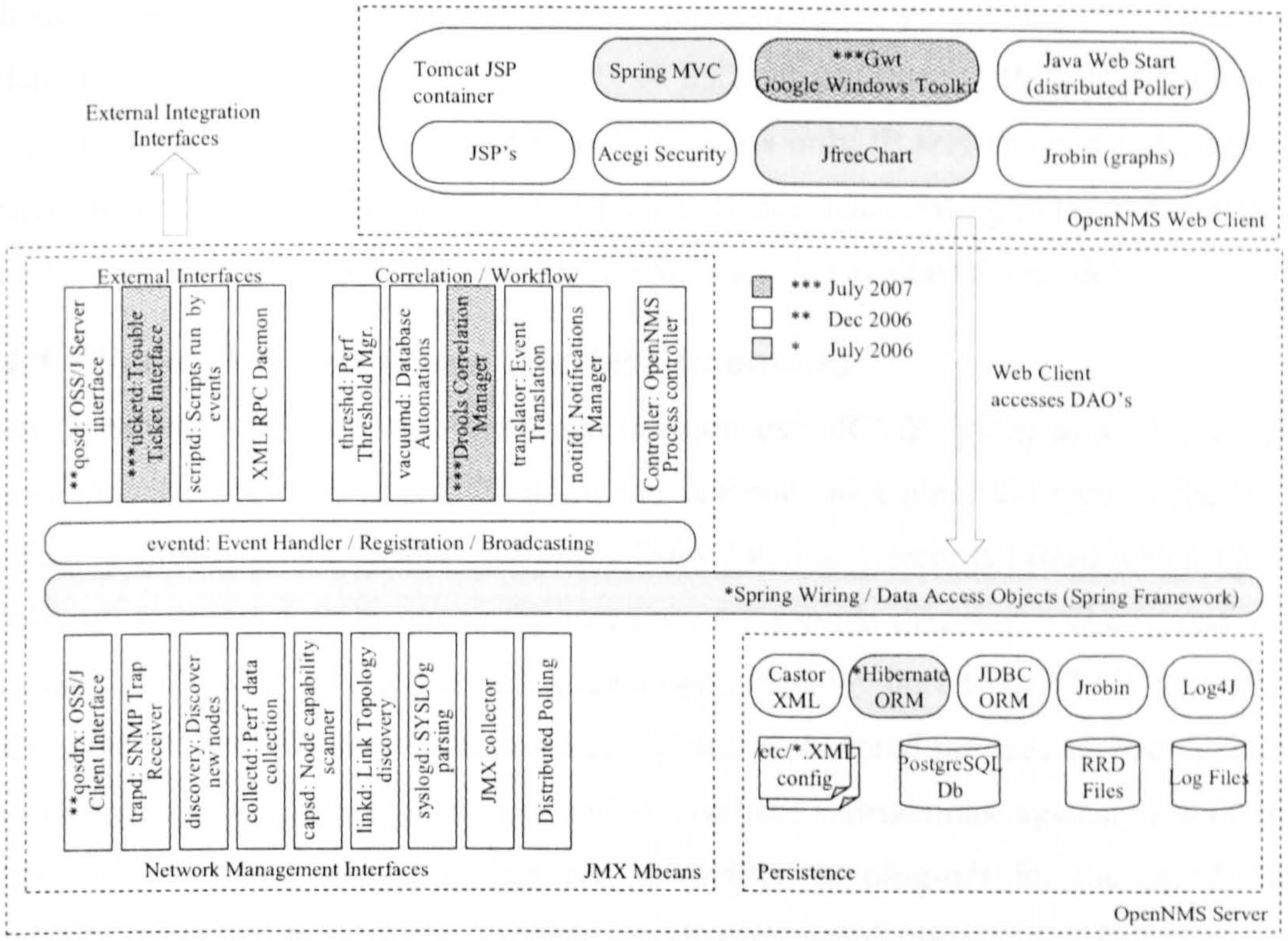


Figure 34 OpenNMS Architecture

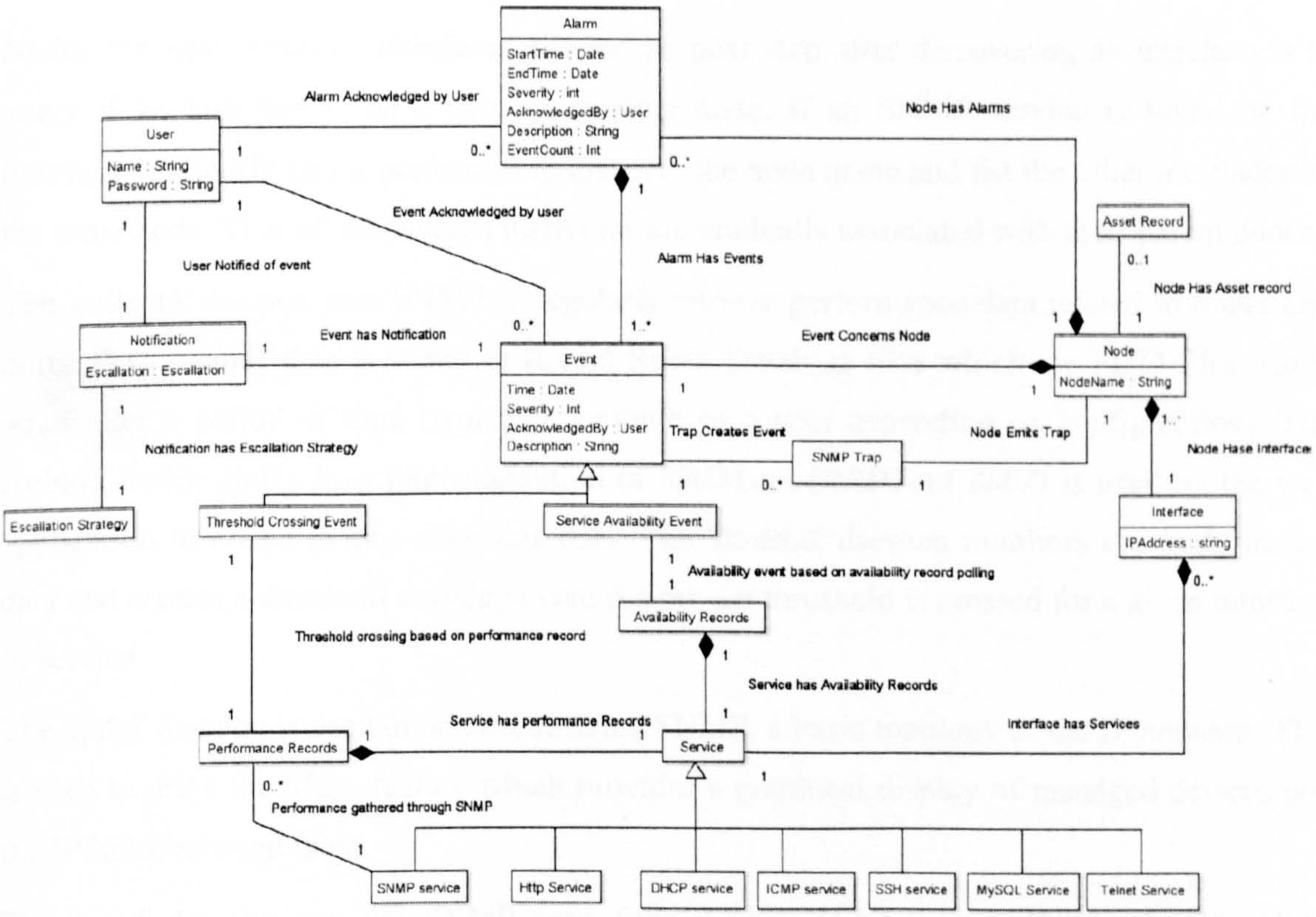


Figure 35 Simplified OpenNMS Object Model

The OpenNMS Daemons have been grouped in Figure 34 as those implementing Network Management interfaces, those implementing Correlation and Work Flow and those Implementing External Interfaces. OpenNMS is event driven and each daemon registers with the eventd daemon for the Events they wish to publish or receive.

Although OpenNMS does not have an explicit object model, Figure 35 has been constructed to explain the relationship between the types of data which OpenNMS handles. OpenNMS's service model is very IP layer 3 centric and considers only IP services and not other layer 2 interface technologies. Although events and performance data relating to layer 2 interfaces can still be collected they are not related to OpenNMS's service availability model.

E.4.1.1 Network Management Interface Daemons

When OpenNMS starts up, the 'discovery' daemon uses ICMP (ping) to scan a configured address range for new interfaces. If an address responds to a ping, the system checks if this interface is already in the database. If not, a 'New Interface' event is raised which the 'capsd' (capability) daemon uses to queue the interface for service discovery. The 'capsd' daemon scans the interface for services at well known ports (SNMP: port 161, HTTP: ports 80,8080, DHCP: port 67, MySQL: port 3306 etc.) and registers discovered services against the interface. Services are monitored by regularly attempting 'synthetic transactions' against each discovered service and additional service pollers can be written as plug-in's for the capsd daemon. OpenNMS continues to scan the services on all discovered ports and a failure to respond generates an 'Outage Event' against a service. This is used to calculate the availability of the services.

Nodes can have multiple interfaces and so the next step after discovering an interface is to query if an interface is on a new or existing node. If an SNMP service is found on the interface, an SNMP Get is performed to discover the node name and list the other interfaces on the same node. Thus all discovered interfaces are gradually associated with their parent nodes.

The 'collectd' daemon uses SNMP to regularly retrieve performance data related to nodes and ports. Performance data is stored in Round Robin Database files which are FIFO files which wrap after a period of time (typically a month or a year depending on configuration). The Jrobin (*Jrobin 2007*) Java implementation of RRDTool (*RRDTool 2007*) is used by the web application to create graphs from this data. The 'threshd' daemon monitors the performance data and creates a threshold crossing event if a pre-set threshold is crossed for a given interface or service.

The 'linkd' daemon is used to determine using SNMP, a basic topology of the IP network. This is used to drive the Maps feature which provides a graphical display of managed devices and the IP links between them.

The 'trapd' daemon receives SNMP traps from devices and translates these into OpenNMS Events using the varbinds and oids to determine the type, severity and textual content of the Event which is generated.

Although OpenNMS is mainly SNMP centric, other monitoring interfaces have been contributed to the project. OpenNMS can parse ASCII Syslog notifications into events using the syslogd daemon and it can monitor the performance of java applications using a JMX

interface implemented by the JMX collector daemon. The qosdrx daemon created by the researcher implements an interface to receive OSS/J alarm notifications.

A recent feature introduced to OpenNMS is the distributed poller framework developed for Kaiser Permanente (*Balog, T. 2007b*). This was developed to allow locations remote to OpenNMS to monitor the availability of services at those locations and report back their status to a central OpenNMS system. Enterprises can use this to monitor service availability at sub offices. The remote pollers are designed to be very simple to install on a PC by pointing Java Web Start (*JCP 2004b*) at a URL hosted in the OpenNMS web app. The remote poller is downloaded, started and registered with the OpenNMS system with minimal user intervention.

E.4.1.2 Correlation and Work Flow Daemons

A number of daemons provide basic correlation and work flow capabilities for OpenNMS.

The 'translator' daemon is used to translate one Event type into another Event type. This can be useful if devices use different traps to generate the same type of indication and the user only wants to see a single Event type for all devices. The 'scriptd' daemon can run user defined scripts in response to a particular Event. This can be used for driving an audio alert or interfacing to another system.

Prior to 2005, OpenNMS could only handle Events but since version 3.1, OpenNMS has had the capability of correlating Events to Alarms. Each Event can be designated as an Alarm Raising or Alarm Clearing Event for a particular type of Alarm. A 'Raise' Event from a Node creates an Alarm against that Node. If more than one 'Raise' Event is generated by a Node, OpenNMS increments an Event count against the Alarm instance. If a 'Clear' Event arrives the Alarm instance is cleared. The introduction of the Alarms feature required modifications to the 'eventd' and 'trapd' daemons and the introduction of a 'vacuumd' daemon which is used to run user defined 'automations' on the database which perform simple correlations of Alarms or Events and clean up old data.

OpenNMS can have many registered users, each with their own username and password. Events and Alarms can be acknowledged by users. Events can also be configured to create 'Notifications' which are escalated between Users according to a pre-defined escalation work-flow strategy if they are not acknowledged within a prescribed period. Notifications can also drive scripts which notify users using a pager or in other ways indicate a significant event needs handled. The 'notifyd' daemon manages the notification escalation process.

OpenNMS has just introduced a correlation daemon in 2007 which uses the Drools correlation engine (*JBoss 2007b*) to correlate events. Matt Brozowski's decision to investigate the use of Drools was influenced by the early use of Drools in the OpenOSS Catalyst and its use in the DVB-T proof of concept. The evolution of this capability is still work in progress.

E.4.1.3 External Interfaces

OpenNMS was initially designed as a stand alone network management application. However an increasing number of users are requiring integration with other systems.

Historically, OpenNMS has not had a published API for integration. A number of users appear to have overcome this by directly reading the SQL database or performance RRD files as a simple form of integration which is facilitated by the fact that they can review the code to understand the function of each of the tables. Integrating directly at the database level is risky as the project may change the database schema in later releases. However being able to see the code and influence the project means that user can manage this risk. Originally OpenNMS used the C version of RRDTool but migrated to the java based Jrobin project which had a slightly different RDD file format. The OpenNMS project has continued support for the C version of RRDTool in order to continue to support customers who have integrated other systems using this method[73].

The first integration interface was provided by the 'scriptd' daemon which can run user defined scripts in response to OpenNMS events. However an increasing number of users have been are looking for more formal integration interfaces. The adoption of Spring (see Appendix E.4.1.5) has greatly assisted in providing a more consistent internal API on which external interfaces can build. The first contribution to take advantage of this was the 'xmlrpc' daemon which provides an XML-RPC interface (*Winer, D. 1999*) by which an external system can consistently populate the OpenNMS database with new nodes, interfaces and services for polling. The 'ttickedd' daemon provides an interface for integrating OpenNMS with a trouble ticket system. Work was started at the 2007 Dev-Jam to create a Representational State Transfer (REST) interface (*Fielding, T. 2000*) for accessing OpenNMS data using HTTP. The OSS/J interface created by this project also leverages Spring.

E.4.1.4 Persistent Storage

OpenNMS stores persistent data in a PostgreSQL database (*PostgreSQL 2007*). Prior to July 2006, OpenNMS used only JDBC to access the database. The implementation was complex and difficult to maintain because database access code spread throughout the web application and the daemons had to be aligned each time the database schema was modified. To significantly ease this problem, the team decided in 2006 to adopt Hibernate (*Hibernate 2007*) as an Object Relational Mapping framework. This had the potential to greatly simplify the problem of schema maintenance, caching, and migrating to databases other than PostgreSQL if desired. However migrating to Hibernate needed to be gradually phased into the project while maintaining the integrity of the application.

---

[73] e.g. Alejandro Galue of Sync Consulting in Venezuela uses RRD files to integrate OpenNMS with an SLA management system - Conversation at OpenNMS Dev Jam 2007.

E.4.1.5 Spring Framework

A major enhancement gradually introduced following the Dev-Jam in July 2006 was the use of the Spring Framework (*Johnson, R. 2005*), (*Spring 2007b*) to provide dependency injection for each of the daemons and web apps. Spring dependency injection allows classes in an application to be 'loosely coupled' by using an XML file to define an 'ApplicationContext' which 'wires' the dependencies between the classes at run time. Instead of hard coding a class's dependencies (using 'new'), the class programmer uses setter methods to allow Spring to pass in implementations of interfaces at run time. This is very useful in open source development as several implementations of the interface (production, beta and prototype) can co-exist in the code base. The Spring wiring determines which one is actually used in the working application.

Spring allowed the team to gradually introduce data access objects (DAO's) and interfaces to the OpenNMS daemons which could be implemented using JDBC or Hibernate. The existing JDBC access methods for all of the daemons were first migrated to Spring so that the daemons were wired to DAO's which referenced the existing JDBC implementations. This was a relatively low risk exercise. In the second stage, the JDBC DAO implementations have been gradually replaced by Hibernate implementations of the same DAO objects. By this method, the introduction of Hibernate has been gradual and aligned with the introduction of new functionality. When work is planned to upgrade a daemon, the Hibernate migration is also done. Any new developments have only used Hibernate. Within a year the full migration should be complete. Thus a significant infrastructure upgrade has been performed without destabilising the project or significantly slowing down the rate of application innovation.

E.4.2 OpenNMS Web Application

Very early versions of OpenNMS (1.n Versions) had a difficult to maintain thick client GUI (Figure 36) based on Swing (*SUN Microsystems 2007*). The Swing GUI was later dropped in favour of a web based UI which is hosted in a Tomcat server. Prior to the introduction of Spring, the web application was complex and also difficult to maintain, consisting of a collection of JSP's and servlets which directly accessed the OpenNMS database.

The introduction of Spring has allowed OpenNMS to take advantage of the Spring Model View Controller (MVC) architecture (*Ladd, S., Davison, D., Devijver, S., & Yates, C. 2006*) which has made the web app design much more consistent. Under the MVC paradigm, an incoming HTTP request is processed by the Controller which selects which JSP should render the result. Previously the JSP's contained application logic but now they merely render data provided by the model classes which access the same DAO's as the server application. Jrobin is used to render RRD files as graphs and JfreeChart is used to render numeric data from the database as graphs. As with the server, Spring migration of the web app has happened

gradually as existing JSP's have been replaced by the MVC implementations. All new pages use Spring MVC.

At the 2007 Dev-Jam, the Google Web Toolkit (GWT) (*Hanson, R. et al. 2007*), (*Google 2007*) was introduced and used to build a 'dashboard' presentation of network status for OpenNMS. GWT will allow the OpenNMS application to re-implement some of the real time capabilities of a Swing GUI by using Ajax[74] java scripts running in the browser to implement graphical widgets. As discussed in Appendix E.5.3, the project is now working on a re-implementation of the Maps display in order to use GWT. OpenNMS are also working on an option to allow the OpenNMS server to directly run a daemon which hosts the web application in Jetty (*Jetty 2007*) as an alternative to a separate Tomcat installation. This would simplify the installation of OpenNMS for new users.

E.4.3 OpenNMS Development Environment and Process

E.4.3.1 Test Driven Development

Spring introduction has allowed OpenNMS to automate Junit (*Junit 2007*) testing of OpenNMS components using Mock Objects (*EasyMock 2007*). Each code package now has a related test suit which uses mock test implementations of dependant objects representing a known data set. Spring wiring is used to wire the class under test into the test environment where the Junit tests are run. The tests are run automatically by the Maven build system every time OpenNMS is compiled.

This has allowed the OpenNMS team to increasingly adopt the 'Test Driven Development' paradigm (*Beck, K. 2003*) where tests are written before the application code and the code is compete when all the tests pass. Test driven development has also been used to help re-factor the original OpenNMS code which was created by developers who have left the project. Tests were written to exercise the key features of the old code and in this way the key dependencies and requirements that were not well understood or documented were extracted. The replacement code was then written to pass the same tests which ensured that the developer didn't miss a key dependency from another part of OpenNMS.

E.4.3.2 Maven 2 project build system

OpenNMS integrates a large ensemble of other open source components to create the application. Fifty five of these sub-components are listed in Table 50. A major headache for OpenNMS has been the version control and distribution of the JARs used to build the project.

The original build system used the Ant build system (*Apache 2007c*) which recursively called Ant builds for each of the OpenNMS modules. Late in 2005, OpenNMS decided to migrate to the Maven 2 build system (*Apache 2007e*), (*Marssol, V. & van Zyl, J. 2006*) as a replacement

---

[74] Ajax (Asynchronous JavaScript and XML) http://en.wikipedia.org/wiki/Ajax_(programming) [Accesed Sept 2007]

for Ant. Maven uses a number of public repositories to store the JARs associated with each release of an open source project. These can be production releases or snapshot releases of the latest development stream. Many open source projects now contribute their releases to the Maven repositories and a number provide snapshots from their automated build systems.

A Maven project is described using a pom.xml file. This lists the project's dependencies and the steps needed to build the project. When a build is performed, Maven downloads the dependant JARs to a repository on the local machine and uses them to build the project. It can also do automated Junit tests as part of the build. This greatly simplifies a developer's job of tracking down and upgrading the project's dependencies. Maven also handles the creation of different RPM packages for Linux distributions and the rendering of project documentation using Docbook (*Docbook 2007*).

OpenNMS is one of the biggest projects to be built by Maven which normally is used for building smaller monolithic components. Migrating OpenNMS to Maven has pushed the boundaries of what Maven can do and has required a number of ingenious 'work-arounds' to be created by OpenNMS. This experience has been fed back to improve the Maven project.

Table 50 Open source components used by OpenNMS

Name	Web Site	Description
batik	http://xmlgraphics.apache.org/batik/	Batik SVG Toolkit
bsf	http://jakarta.apache.org/bsf/	Bean Scripting Framework (BSF)
bsh	http://www.beanshell.org/	JSR-274 the BeanShell Scripting Language
cglib	http://cglib.sourceforge.net/	cglib Java Code Generation Library
c3p0	http://sourceforge.net/projects/c3p0	c3p0 is an easy-to-use library for augmenting traditional (DriverManager-based) JDBC drivers with JNDI-bindable DataSources
castor	http://www.castor.org/	Castor provides Java-to-XML binding,
commons-beanutils	http://commons.apache.org/beanutils/	BeanUtils component provides easy-to-use wrappers around Reflection and Introspection APIs
commons-dbcp	http://commons.apache.org/dbcp/	Database Connection Pools
commons-httpclient	http://commons.apache.org/httpclient/	Feature-rich package implementing the client side of the most recent HTTP standards and recommendations
commons-io	http://commons.apache.org/io/	Library of utilities to assist with developing IO functionality
commons-lang	http://commons.apache.org/lang/	Lang Component provides a host of helper utilities for the java.lang API
commons-logging	http://commons.apache.org/logging/	Bridge between different logging implementations
concurrent	http://g.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html	Standardised, efficient versions of utility classes commonly encountered in concurrent Java programming (replaced as standard in J2SE5)
json-lib	http://json-lib.sourceforge.net/	JSON (JavaScript Object Notation). JSON-lib is a java library for transforming beans, maps, collections, java arrays and XML to JSON and back again to beans and DynaBeans
easymock	http://www.easymock.org/	EasyMock provides Mock Objects for interfaces in JUnit tests by generating them on the fly using Java's proxy mechanism
extremecomponents	http://www.extremecomponents.org/extremesite/index.jsp	eXtremeComponents are a series of open source JSP custom tags that render advanced displays
gwt-widgets	http://gwt-widget.sourceforge.net/	GWT Server Library is a collection of Java server side components for the Google Web Toolkit AJAX framework
groovy	http://groovy.codehaus.org/	Is an agile and dynamic language for the Java Virtual Machine
httpunit	http://httpunit.sourceforge.net/	HttpUnit emulates the relevant portions of browser behaviour for unit testing
jakarta-regexp	http://jakarta.apache.org/regexp/	Java Regular Expression package
javax.activation	http://java.sun.com/j2ee/sdk_1.3/techdocs/api/javax/activation/package-summary.html	The JavaBeans Activation Framework is used by the JavaMail(TM) API to manage MIME data
javax.jnlp	http://java.sun.com/j2se/1.5.0/docs/guide/javaws/jnlp/javax/jnlp/package-tree.html	Java Network Launching Protocol (JNLP)

javax.mail	http://java.sun.com/products/javamail/javadocs/javax/mail/package-summary.html	JavaMail API
javax.persistence	http://java.sun.com/javaee/5/docs/api/javax/persistence/package-tree.html	The javax.persistence package contains the classes and interfaces that define the contracts between a persistence provider and the managed classes and the clients of the Java Persistence API
jnlp-servlet	http://java.sun.com/j2se/1.4.2/docs/guide/jws/downloadservletguide.html	Servlet in the developer's pack that can be used to package a JNLP file and its associated resources in a Web Archive (.war) file
jstl	http://java.sun.com/products/jsp/jstl/	JavaServer Pages Standard Tag Library
javax.servlet	http://tomcat.apache.org/tomcat-5.0-doc/servletapi/javax/servlet/http/package-summary.html	The javax.servlet.http package contains a number of classes and interfaces that describe and define the contracts between a servlet class running under the HTTP protocol and the runtime environment provided for an instance of such a class by a conforming servlet container
org.mortbay.jetty	http://www.mortbay.org/	Jetty is an open-source, standards-based, full-featured web server implemented
jfreechart	http://www.jfree.org/jfreechart/	Java chart library that makes it easy for developers to display professional quality charts in their applications
smack (jivesoftware) smackx	http://www.jivesoftware.com/builds/docs/smack/latest/javadoc/	Open Source XMPP client library Extensible Messaging and Presence Protoco
jmock	http://www.jmock.org/	jMock is a library that supports test-driven development of Java2 code with mock objects
jrobin	http://www.jrobin.org/index.php/Main_Page	JRobin, a Java port of RRDTool
junit	http://www.junit.org/index.htm	JUnit is a regression testing framework
jwebunit	http://jwebunit.sourceforge.net/	JWebUnit is a Java framework that facilitates creation of acceptance tests for web applications
log4j	http://logging.apache.org/log4j/docs/index.html	Java Logging package
mx4j	http://mx4j.sourceforge.net/	MX4J is an Open Source implementation of the Java(TM) Management Extensions (JMX) and of the JMX Remote API (JSR 160) specifications
net.sf.jradius	http://jradius.org/	JRadius, the open-source Java RADIUS processing server and Java RADIUS Client.
openldap jldap	http://www.openldap.org/jldap/	LDAP Class Libraries for Java (JLDAP)
org.acegisecurity	http://www.acegisecurity.org/	Acegi Security is a powerful, flexible security solution for enterprise software, with a particular emphasis on applications that use Spring
org.springframework	http://www.springframework.org/	Spring Framework, the leading full-stack Java/JEE application framework
org.dhcp.jdhcp	http://www.dhcp.org/javadhcp/	The goal of JDHCP is to add Dynamic Host Configuration Protocol (as specified in RFC 2131) handling to Java
org.hibernate	http://www.hibernate.org/	Object/relational persistence and query service.
org.samba.jcifs	http://jcifs.samba.org/	JCIFS is an Open Source client library that implements the CIFS/SMB networking protocol in 100% Java. CIFS is the standard file sharing protocol on the Microsoft Windows platform (e.g. Map Network Drive).
org.snmp4j	http://www.snmp4j.org/	SNMP4J - The Object Oriented SNMP API for Java Managers and Agents
oro	http://jakarta.apache.org/oro/	The Jakarta-ORO Java classes are a set of text-processing Java classes that provide Perl5 compatible regular expressions,
poi	http://poi.apache.org/	Apache POI - Java API To Access Microsoft Format Files
postgresql	http://www.postgresql.org/	PostgreSQL is a powerful, open source relational database system. It has more than 15 years of active development
quartz	http://sourceforge.net/projects/quartz/	An enterprise-class job scheduler for integration with stand-alone java applications and full-scale J2EE applications
RRDTool	http://oss.oetiker.ch/rrdtool/index.en.html	RRDtool is the OpenSource industry standard, high performance data logging and graphing system for time series data
struts	http://struts.apache.org/	Apache Struts is a free open-source framework for creating Java web applications
standard taglibs	http://jakarta.apache.org/taglibs/doc/standard-doc/intro.html	Standard Taglib 1.1, an implementation of the JSP Standard Tag Library (JSTL),
tomcat catalina	http://tomcat.apache.org/	Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and JavaServer Pages technologies
xerces	http://xerces.apache.org/xerces-j/	The Xerces Java Parser 1.4.4 supports the XML 1.0 recommendation and contains advanced parser functionality
jep	http://www.singularsys.com/jep/	JEP - Java Math Expression Parser JEP is a Java library for parsing and evaluating mathematical expressions.

E.4.3.3 Code Management and Developer Communications

Most OpenNMS developers use Eclipse (*Eclipse 2007a*) for development. Among other things, this can accept plug-in Spring IDE tools for working with Spring (*Spring 2007a*). All OpenNMS code is hosted on Sourceforge[75]. Originally OpenNMS used CVS (*CVS 2007*) to moderate the concurrent work of the developers in the project (*Fogel, K. & Bar, M. 2001*). However at the same time as adopting Maven 2, OpenNMS also migrated their code base on Sourceforge from CVS to Subversion (*Subversion 2007*) which allows much finer control of which version of OpenNMS a developers is checking out. OpenNMS have also introduced a continuous build process using CruiseControl (*CruseControl 2007*) which builds the project each time a commit is made to the subversion repository[76]. This, combined with Maven and extensive test coverage, allows the project to use continuous integration (*Fowler, M. 2006*).

Online documentation is hosted on the OpenNMS wiki[77] and bug tracking is done using bugzilla[78]. Several mail exploders are used to communicate within the project[79]. The core developers also keep in touch with an OpenNMS IRC channel which is used for friendly banter as much as for serious work.

E.5 Innovation in the OpenNMS Project

As observed in Appendix E.3, all of the core developers have had prior experience of the major network management platforms and of running large networks. OpenNMS does not have a written design specification but the prior experience of the developers has certainly given them a common understanding of the types of features they should be targeting. At a superficial level, one could dismiss the idea of OpenNMS enabling innovation by suggesting that it is merely re-implementing network management techniques common in other established COTS OSS products. Certainly in its early days, OpenNMS needed to lay a foundation of basic capabilities in order to compete in its target market. However when one looks more deeply, there is a high level of innovation occurring in the project which is more subtle than merely copying existing features in other products.

E.5.1 Innovation in the OpenNMS Business Model

The most noticeable innovation is in learning how to use open source in a successful network management business model. This learning is happening across the project in both the OpenNMS Group (and their partners) who are providing OpenNMS services and in the companies who are adopting OpenNMS into their infrastructure.

---

[75] OpenNMS Sourceforge Code Repository http://sourceforge.net/projects/opennms/

[76] OpenNMS Continuous Integration Results http://internal.opennms.com:8080/

[77] http://www.opennms.org/index.php/Main_Page [Accessed Sept 2007]

[78] http://bugzilla.opennms.org/ [Accessed Sept 2007]

[79] OpenNMS Lists; announce (Announcements about new releases and other important info. Very low traffic); cvs (A list to receive notifications of commits to the source-code repository); discuss (General discussion list. Your first port of call for non-paid support).; install (Discussion list for installation specific issues); devel (Discussion list for those involved in, or wanting to be involved in, development of OpenNMS documentation); [[Accessed Sept 2007] http://www.opennms.org/index.php/Mailing_List_FAQ#OpenNMS_Mailing_Lists

Firstly, an OpenNMS champion in each user organisation has had to build the business case to persuade their management that an open source solution could meet their needs in the face of the prevailing fear, uncertainty and doubt promulgated by the COTS vendors. In several cases, OpenNMS has been chosen in the context of direct bids from competing products such as HP OpenView (*HP 2007b*).

Secondly, the adopting organisations have had to make decisions about the level of external support they required vs. the level of expertise they needed to grow internally. It is very clear that every company has drawn this line in a different place. In some cases, organisations have got this wrong and not allocated sufficiently skilled staff or time to the product. These have represented problem customers for the OpenNMS Group. In other cases, companies have been content to use the product 'out of the box' with no customisation and a minimal support contract. In many cases staff have volunteered their own time to the project, seeing this as a way to enrich their own skills and many companies have actively allowed staff time to contribute or received training.

Thirdly, several companies such as RackSpace, New Edge Networks and Swisscom have used OpenNMS in business critical applications and paid for significant enhancements to OpenNMS in order for it to integrate into their environments. Those organisations that have made contributions or paid for custom developments have needed to understand more clearly where their competitive advantage lies and whether this would be affected by contributing OpenNMS developments back to the community. This consideration has had to be weighed against the difficulty and cost of obtaining the features in any other way (i.e. internal development or the modification of COTS products).

The OpenNMS group have needed to learn flexibility in how they offer services and work with members of the OpenNMS community. Their approach towards the community is essentially very generous. They maintain the core product and documentation wiki and do what they can to help community members make a contribution, whether or not they are paying customers. Paying customers obviously get priority when it comes to support but the OpenNMS Group also contribute to the free support through the mail lists - although they tend to hold back to encourage other community members to step in first. However since new users may eventually become customers, the OpenNMS Group pay particular attention to servicing the 'install' list in order to help people evaluate the product and not give up because they cannot get it working.

As can be seen, the business model requires a different approach to the traditional 'shrink wrapped' software sale. OpenNMS Group have tried unsuccessfully in the past to use sales professionals to promote the product and have come to realise that most of their sales have arisen following a period of technical engagement to help the customer see where and how OpenNMS could solve their problem. The offering of 'free' consultancy to attract business has to be carefully balanced against paid for consultancy which is their bread and butter.

E.5.2 Innovation in core technologies

As can be see from the description of the product in Appendix E.4, OpenNMS has been highly innovative in its adoption of leading edge open source implementation technologies such as Maven, Spring and latterly GWT. Furthermore, the project continuously integrates with the latest 'snapshot' version of its dependencies and so automatically leverages the latest developments within its contributing projects. The move to these new frameworks has made it increasingly possible for external developers to contribute features and for the project to accept contributions. The use of leading edge technologies also makes the project more attractive to developers who want to work at the cutting edge.

A major technology refresh has occurred during the last two years. Although COTS products do periodically refresh their technology, this is usually aligned with a major release cycle and timed with a chargeable customer upgrade. In OpenNMS, users have complete choice over which version they stick with or if they wish to evaluate the latest developments in the Trunk of the code base.

In a COTS product, the road map is determined in advance of any development through a New Product Introduction process as described in Section 3.6.1. However, OpenNMS have found that their roadmap is very much determined by the interests of their varied user community. Although the core OpenNMS project has its own nominal roadmap, OpenNMS has increasingly had to learn how to accept and integrate developments which have been contributed by external parties without prior warning.

The pattern appears to be that OpenNMS is adopted by a user for a particular project and modified or extended without significant reference back to the core project. Later the user contacts the project and offers the feature to the community. The pattern for accepting innovations is generally to create a branch of the source tree where the innovation is integrated into a particular version of OpenNMS and made available to the community. Later when the innovation is stable, it is accepted into the core code. This is what happened in the JMX project (Appendix E.5.5), the Maps project (Appendix E.5.3) and in the OSS/J contribution (Section 6.4).

The following section will consider how user led innovation has enhanced OpenNMS capabilities and where OpenNMS has contributed to Academic research.

E.5.3 OpenNMS Maps Innovation

In Network management systems, maps are used to show graphically the topology and state of elements in a network. Originally OpenNMS was planned to have a Maps feature built into the Swing GUI as illustrated in Figure 36 (*Giles, S. 2000*). This feature was dropped when OpenNMS moved to a web GUI but a number of database table entries referring to Maps remained. The justification given by Tarus for not having maps was that while most network

management systems have a map capability, users often find in practice that it is impractical to create a map for very large networks (*Balog, T. 2007a*).

However user interest in a Maps feature persisted and a community member, Derek Glidden created a simple Scalable Vector Graphics (SVG) (*W3C 2003*) based Maps feature in 2002 which he hosted on a separate site[80]. Unfortunately this could only display on Microsoft Internet Explorer using an SVG plug-in from Adobe (*Adobe 2007*) and could only be rendered on a small number of nodes. The Maps could not be rendered on Firefox (*Mozilla 2007*) and so were unattractive for a Linux environment. Nevertheless, some users found the feature useful and it was eventually incorporated into the OpenNMS code tree but by default switched off because of the scalability issues.

Between 2004 and 2005, Antonio Russo's consultancy company sold a solution using OpenNMS to a customer in Italy. As part of the solution, he was required to re-implement the MAP functionality with more features (Figure 37). After developing this feature, Antonio approached OpenNMS to see if it could be incorporated into the core. As a first step separate branch called 'Italian Adventures' was created so that the code could be evaluated. Antonio attended the 2006 Dev-Jam and subsequently modified his code to use the new Spring framework and Maven. The code was then incorporated as an option in the OpenNMS trunk. Unfortunately the code still had the limitation of only being viewable on IE and so at Dev-Jam 2007, the process was begun to port the code into the GWT framework and overcome this limitation.

It can thus be observed how innovation has happened independently of the main project and been subsequently incorporated into the core. Furthermore, ideas which were partly implemented by one party have been picked up and improved by other parties as time has gone along. Finally, one should note that the Maps feature was clearly not a priority for the core team but was supported by some users using their own resources and eventually picked up by the main project. This could not normally happen in a COTS product.

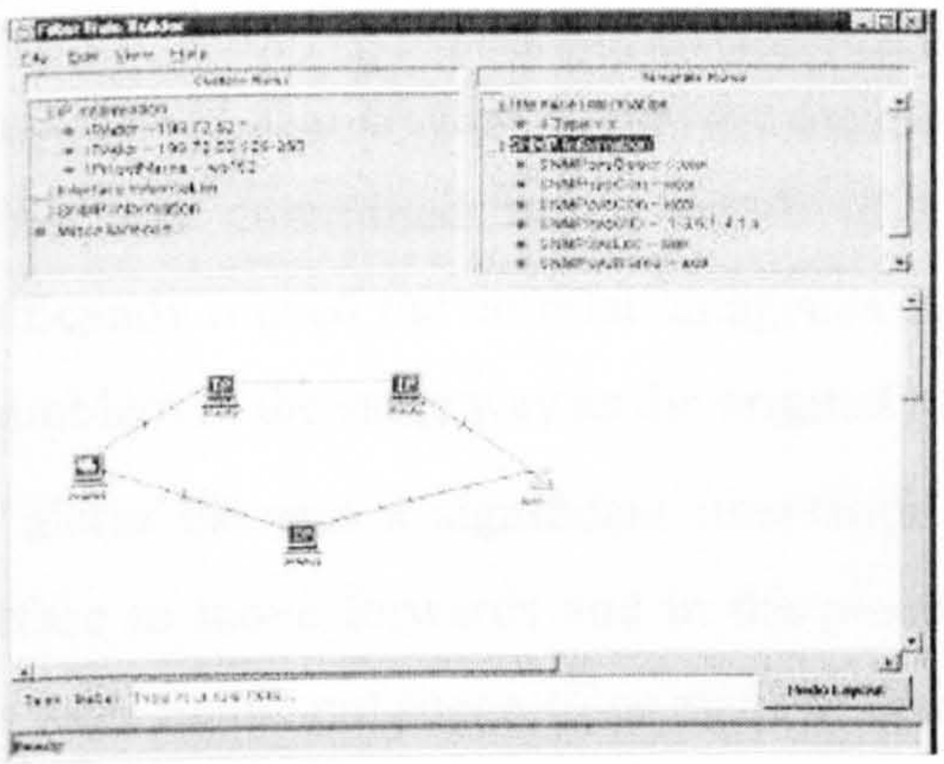


Figure 36 Original OpenNMS 1.0 Swing GUI

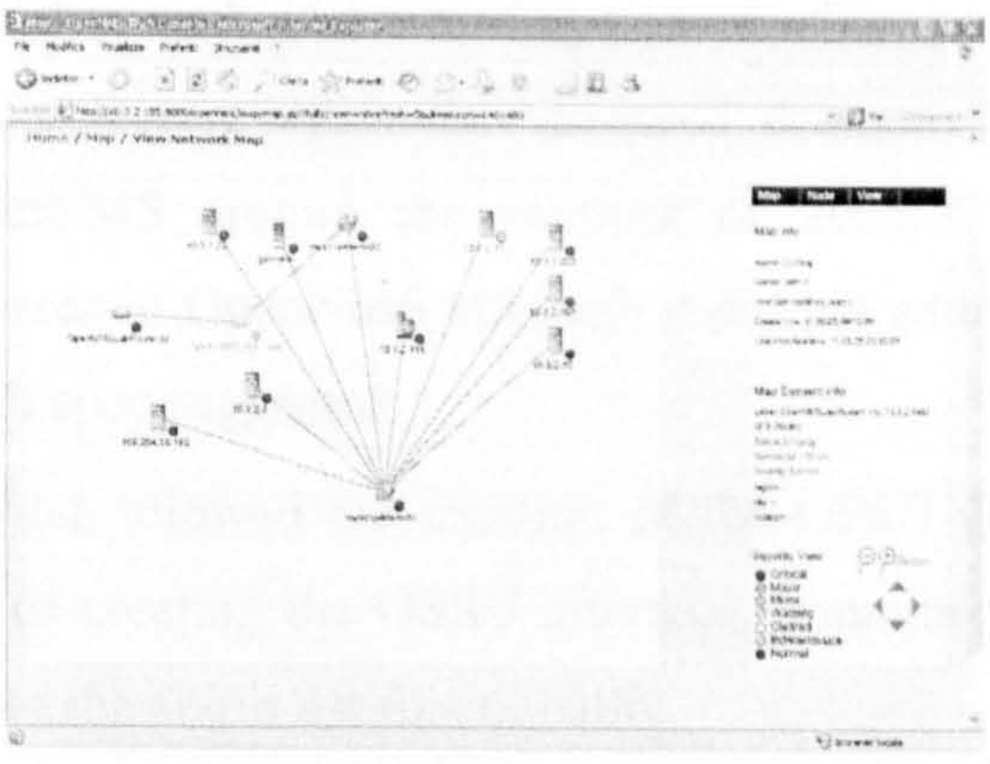


Figure 37 OpenNMS Italian Adventures SVG GUI

[80] http://projects.nksi.com/openmms-svg-map/ - now defunct

E.5.4 OpenNMS Correlation Innovation

From the outset, OpenNMS has had the ambitious objective of creating a comprehensive open source event correllator. The original project was called the Malevolent Alarm Judication Inference (MAJI) and the original draft specification was produced by an external contributor, Douglas Stevenson (*Stevenson, D. W. 2000*). The MAJI project attracted some attention from other researchers in the field and the specification was considered by Vaarandi, before he went on to create his own open source correlation engine, SEC[81] (*Vaarandi, R. 2002b*), (*Vaarandi, R. 2002a*)

Stevenson collaborated with Rick Casey, a student who was doing an MSc project on event correlation at the University of Colorado. Shane O' Donnell's weekly report notes that Rick and Doug were working jointly on the specification during Rick's holidays but noted that they still had more to do and that 'correlation is a daunting task' (*O'Donnell, S. 2001a*). However since progress was slow, OpenNMS would have to make do with 'rudimentary event correlation' which;

> '..includes a hook in eventd to watch for certain events and either A) suppress duplicate events or B) watch for events that cancel out the original events. We're referring to these internally as "event singlification" and "event cancellation". These are both simple approaches to buy us time until we can get MAJI assembled'. (*O'Donnell, S. 2001b*).

MAJI was never completed although Casey eventually produced a prototype event correllator using Perl for his own purposes and acknowledges the help given by the OpenNMS team at the end of his MSc thesis (*Casey, H. R. 2002*).

When the OpenOSS Catalyst was being implemented at the beginning of 2005 (Section 4.2), OpenNMS had not gone much further with event correlation and the catalyst had to integrate with OpenNMS using Events because it had no correlated alarm capability. UoS Student Tim Griffiths created a very simple correlation prototype using the Drools (now JBoss rules) (*JBoss 2007b*) inference engine for the Catalyst.

At the same time, David Hustace of OpenNMS Group began work on an Alarm Correlation feature for OpenNMS which allowed events to be tagged as alarm raising or alarm clearing Events and drive the contents of a separate Alarm list, each Alarm having a count of its raising Events. David also created a database automations feature within the 'vacuumd' daemon. The feature was committed to the trunk of OpenNMS around the summer of 2005. This significantly moved the correlation agenda forwards in OpenNMS although it did not address the problem in the same way as the original MAJI spec suggested.

The alarm list was a significant innovation which allowed the creation of the OSS/J QoS interface to move forwards and in the process of creating the OSS/J interface, a number of bugs were found and suggestions made to improve the alarm list functionality.

At the beginning of 2007, Matt Brozowski had another correlation problem to solve for a customer and, influenced by the use of Drools (JBoss Rules) (*JBoss 2007b*) in the OpenOSS

Catalyst and Sidonis StateWise, decided to try and use Drools within OpenNMS to do this task. He then implemented the OpenNMS correlation engine as a first experiment in building a rules engine into the system. Having done a first integration of Drools with OpenNMS, the way is now clear to investigate how this rules engine can be better used to solve the event correlation problem.

This case study illustrates how OpenNMS has leveraged collaborative innovation throughout its life. Some projects have been tried and dropped while other projects have succeeded. However throughout the process, the fact that OpenNMS is open and accessible has meant that different parties could experiment until a workable solution was found.

E.5.5 User Innovation in Network Management around OpenNMS

OpenNMS has a place in a wider eco-system of practitioners, projects and researchers who are working in the Network Management field. Several of these projects have leveraged or contributed back to OpenNMS. A few examples are listed in Table 51.

A number of academic projects have also leveraged or referenced the work being done in OpenNMS. Shihada has used OpenNMS as his template in considering the conceptual and concrete architectures of network management systems in a project delivered as part of his PhD (*Shihada, B. 2002*). Microsoft researchers have examined the OpenNMS topology mapper in considering how to discover the Layer 2 topology of networks (*Black, R., Donnelly, A., & Fournet, C. 2004*). John Hoag at Ohio University has been leveraging the work done with OpenNMS and OpenOSS to propose an architecture for network operations and management based on state and services (*Hoag, J. C. 2006*). Marcus Jaton has lead an extensive project sponsored by the Swiss Tcom institute leveraging OpenNMS to create an Experimental Network Management Platform ((*Jaton, M. 2002*), (*Jaton, M. 2003b*), (*Jaton, M. 2003a*) all in French). Although unfortunately, this does not appear to have delivered any contributions back to OpenNMS.

Table 51 Other Open Source Projects interacting with OpenNMS

JMX Monitoring	Java Management Extensions (JMX) is technology developed under the JCP process by Sun to manage Java applications (*SUN Microsystems 2005b*). Although OpenNMS used JMX internally, the first use of OpenNMS to manage external applications using JMX was documented in 2002 by an IBM developer who published an application note on how to use OpenNMS to manage JMX using a JMX to SNMP converter (*Li, S. 2002*). In 2004, an OpenNMS user, Michael Jamison, took this further by writing a JMX poller as an extension to the collectd daemon. This was a complex task as 'collectd' was one of the less easy to understand parts of OpenNMS. The JMX polling feature was included in OpenNMS 3.1 and quickly applied by Ocado to managing their extensive Tomcat installations. Jonathan Sartin contributed an application note to explain to other users how they had used this feature (*Sartin, J. 2005a*).
JoeSNMP	JoeSNMP was originally created in 2000 by the OpenNMS project as an open source java SNMP library. Dimitris Andreadis used JoeSNMP to create an SNMP agent for JBoss 3.2.2 which intercepted JMX notifications emitted by MBeans, converted them to SNMP traps and sent them to SNMP managers. In JBoss v4.0.4, the snmp-adaptor was extended to support basic get and getNext to allow one to query snmp-oids, which in turn were mapped to JMX attribute gets (*JBoss 2007d*). To facilitate this JoeSNMP was forked from OpenNMS as a

[81] http://simple-evcorr.sourceforge.net/ Simple Event Correlation Engine

	separate project on Sourceforge under the LGPL licence with Dimitris Andreadis as one of the maintainers along with the OpenNMS team so that it could be picked up by other projects needing SNMP capabilities (*JoeSNMP 2007*).
Nagios	OpenNMS has close relationships with several other projects. Nagios (*Nagios 2007*) is a simpler management application than OpenNMS which has had wide adoption for smaller scale monitoring tasks (*Barth, W. 2006*), (*Josephsen, D. 2007*). It has a large number of user contributed plug-in's and also uses RRDTool for saving performance data. Since version 1.2.8 (2006) OpenNMS has included the capability of interrogating Nagios plug-in's and leveraging the work of the Nagios community.
OSSIM	The Spanish Open Source Security Information Management (OSSIM) (*OSSIM 2007*) project can use OpenNMS as a service availability monitor, although as a C based distribution for Linux their preference seems to be to use Nagios.

Appendix F
OpenNMS OSS/J Interface Design

Appendix F OpenNMS OSS/J interface Design

This appendix provides details of the overall development process and design of the OpenNMS OSS/J interface.

F.1 Project Chronology

Unlike the other interventions recorded in this Thesis, the researcher's involvement with the OpenNMS project has been continuous and has had few clearly delineated milestones. The association has gradually deepened throughout the project in support of the other interventions. Table 52 lists (in reverse chronological order) the major OpenOSS development milestones and overlapping OpenNMS development milestones and activities.

Between March and July 2004 (1) an extensive internet survey was performed to identify suitable projects to use as the basis of an early proof of concept of open source in telecoms. OpenNMS and OTRS (*OTRS 2007*) were chosen and agreed with BT for investigation.

By using three sponsored Southampton University students during the summer of 2004 (2) it was possible to consider the issues faced when a collaborative project team unfamiliar with the chosen open source components begins investigating an open source solution. The researcher wrote an outline specification to brief the students on the objective of trying to use an OSS/J TT (Trouble Ticket) interface (*OSS/J 2002a*) to integrate OpenNMS and OTRS. The project could be characterised as a simple Glue-Code COTS Based Application (GCBA) as described in Table 17, Section 3.5.1.1.

Although Christophe Ebro from Sun provided some help in understanding the OSS/J TT interface, the team relied entirely on the project documentation and mail lists for OpenNMS and OTRS with no direct contact to the projects. A development release of OpenNMS 1.1.3 was used since it could be installed from RPM's and didn't require OpenNMS to be compiled or modified by the developers. The OpenOSS project was created on Sourceforge and used as a repository for the work (*OpenOSS 2007*). Apache Ant (*Apache 2007c*) was used as the build system. Until much later, the OpenOSS code was compiled in a separate project and offered to the community as plug-in JARs and configuration files to install on top of a standard OpenNMS installation. (This was a similar approach as that taken with Derek Glidden's early Maps project discussed in Appendix E.5.3).

The OpenNMS group were first contacted directly on the 4th of October 2004 after the team had built a simple trouble ticket simulation. When Dave Hustace was told that the Catalyst intended to demonstrate the simple integration at TM World in Long Beach in October 2004 he volunteered to come out to the conference and support the presentations.

Following TM World in Long Beach, the OpenNMS group agreed to help the Catalyst team integrate an OSS/J Qos interface into OpenNMS for the proposed OpenOSS Catalyst (4). OpenNMS 1.1.5 was used since it had support for Jrobin which the OSS/J Performance

Management interface needed. Again a supported development release was used and not the latest snapshot of the project.

Following the OpenOSS Catalyst, another summer student worked with the researcher to integrate the new OpenNMS alarm list into the OSS/J interface (5). This time a snapshot of OpenNMS 1.3.0 was downloaded and compiled from source. Subsequent to the assistance of this student, the researcher worked alone on further developments.

Whereas the early Catalyst project could be characterised as creating a simple Glue-Code COTS Based Application (GCBA), the work with OpenNMS increasingly became more like a Non-COTS intensive project (Table 17, Section 3.5.1.1); i.e. a traditional development project where new functionality was being added to the core of OpenNMS which leveraged new components such as XMLBeans.

Between September 2005 and January 2006 (6), the researcher worked with Arqiva on the DVB-T proof of concept and developed the interface to Sidonis StateWise (*Sidonis 2007*). The work tracked OpenNMS as it introduced Maven and prepared to introduce Spring by creating Data Access Objects (DAO's).

Until this point, the OSS/J interfaces had only implemented the Java Value Type (JVT) pattern. From January 2006 to March 2006 (6) the open source component XMLBeans (*XMLBeans 2006*) was leveraged to create an XML interface corresponding to the OSS/J XML Value Type pattern (XVT) which ran over a Java Messaging Service (JMS) (*JCP 2003c*) messaging bus provided by JBossMQ (*JBoss 2007c*). The OpenNMS qosdaemon project was created to incorporate the qosd component which published an OSS/J alarm list. Between April 2006 and July 2006 (7) qosdrx was created enabling OpenNMS to receive OSS/J alarms. Extensive Junit test classes are also created as part of the test driven development process.

Between August 2006 and September 2006 (8), the qosdaemon code was adapted to use the Spring Framework (*Spring 2007b*) with JDBC DAO's in OpenNMS. The Ant build system was replaced with the Maven 2 (*Apache 2007e*) build system in preparation for integration with OpenNMS.

From September 2006 to December 2006, (9) the code was migrated into the OpenNMS build system and the JDBC DAO's migrated to Hibernate DAO's. The project was split into two projects; a component which would be contributed to OpenNMS under the GPL licence (qosdaemon) and an OSS/J component, OSSBeans (*OSSbeans 2007*), which would be generally available under the Apache 2 licence. The OSSbeans project was committed to Sourceforge with its own Maven repository so that OpenNMS could download and use the OSSBeans JARs during its Maven build.

The code was finally contributed to OpenNMS in December 2006 and included in the OpenNMS 3.2.3 release on 19 Jan 2007 (10). As a footnote, in recognition of the researcher's

contribution to the project he was awarded with membership of the order of the green polo by the OpenNMS developers n July 2007 (11).

Table 52 OpenOSS development milestones and overlapping OpenNMS milestones

OpenOSS Development Task / Milestones (Note - Working back from the present)	**OpenNMS version/date**	**Dependant OpenNMS Feature/Activity**
(11) Craig Gallen Awarded Membership of the 'Order of the Green Polo' August 2007	July 2007	• Dev-Jam 2007 Third OpenNMS Developers Conference
	1.3.6 30-Jul-2007	• Code cleanup
	1.3.5 09-Jul-2007	• Minor bug fixes
	1.3.3 21-Jun-2007	• Major feature enhancements • Drools correlation • Distributed Polling
(10) 19-Jan-2007 OSS/J code released • qosdaemon integrated into OpenNMS Release	1.3.2 19-Jan-2007	• Major feature enhancements
(9) in July 2007. • Handover OSS/J interface to OpenNMS team; education for OpenNMS developer team and passing over documentation etc • Completed OSS/J interface for OpenNMS • Completed final integration of OpenNMS build with OSS/J interface • Documentation and code on Sourceforge • 'Mavenised' project and incorporated in build system • Integrated qosdaemon into Hibernate Alarm DAO's	30/11/2006 to 3/12/2006	• Handover Meeting with OpenNMS in North Carolina.
	1.3.2 snapshot 16/11/2006	• 1.3.2 snapshot now includes qosdaemon
	28/9/06 to 4/10/06	• Meeting with OpenNMS in North Carolina. Working with Dave Hustace to integrate Alarm List with Hibernate DAO's
	1.3.2 snapshot	• Alarm list moves to Hibernate DAO's
(8) August 2006 - September 2006 Integrated qosdaemon into OpenNMS build • Wire qosdaemon into Spring JDBC Alarm DAO's • Split project into OSSBeans and qosdaemon in separate projects • Built qosdaemon within OpenNMS Maven build process and Maven project	1.3.2 snapshot	• Alarm List uses JDBC DAO's (Not Hibernate) • Spring introduced
	July 2006	• Dev-Jam 2006 Second OpenNMS Developers Conference
(7) April 2006 - July 2006 Development of qosdrx using XML alarms • Using OpenNMS 1.3.2 snapshot build from SVN Trunk • qosdrx can receive alarms and update OpenNMS alarm list • Moved qosdaemon project to subversion and Maven	1.3.2 snapshot	
(6) January 2006 - March 2006 Development of qosd using XML alarms • Using OpenNMS 1.3.2 snapshot build from SVN Trunk • Introduction of XMLBeans and XVT OSS/J messages • qosd sends Alarm list through OSS/J Qos FM XVT and JVT interface to Sidonis StateWise	1.3.2 snapshot	• OpenNMS moves to Subversion (SVN) and Maven build process
(5) September 2005 - January 2006 Initial integration with Sidonis StateWise • Using OpenNMS 1.3.0 snapshot build from CVS Trunk • qosd sends Alarm list through OSS/J Qos FM interface to Sidonis StateWise • JVT interfaces - no XML	1.3.1 22-Nov-2005	• Minor bug fixes
	1.3.0 24-Oct-2005	• Major feature enhancements • Alarms / Automations added
	Oct 2005	• Dev-Jam 2005 First OpenNMS Developers Conference - Maven & DAO introduction (not Spring)
(4) June-September 2005 Initial investigation of OpenNMS Alarm List • Using OpenNMS 1.3.0 snapshot build from CVS Trunk (not a release) • qosd sends Alarm list through OSS/J Qos FM interface • JVT interfaces - no XML • qosdaemon created as separate project - not integrated into OpenNMS build system	1.3.0 snapshot	• Alarms and Automations.
(3)January-May 2005 OpenOSS Catalyst uses OpenNMS 1.1.5 • Separate project build and installation on top of standard OpenNMS • OSS/J Qos FM interface sends Events list not Alarms • OSS/J Qos PM interface needed Jrobin in OpenNMS • JVT interfaces - no XML • JBoss J2EE application server	1.1.5 31-Jan-2005	• Major feature enhancements
	1.1.4 26-Oct-2004	• Jrobin added
(2)August-October 2004 Initial investigation (TMW Long Beach) uses OpenNMS 1.1.3 • OSS/J TT interface between OpenNMS & OTRS • JVT interfaces - no XML • Free Sun JAS J2EE application server	1.1.3 17-July-2004	• Ant Build system added to OpenNMS
(1) March-July 2004 Initial Investigations into OpenNMS suitability	1.1.2 17-March-2004	

F.2 Project Overview

F.2.1 Introduction to OSS/J

Operational Support Systems through Java (OSS/J) is an initiative initially championed by Sun Microsystems but now incorporated into the Telemanagement Forum Prosspcro program (*TMForum 2007g*). OSS/J provides a means to implement the TMForum New Generation Operational Support Systems (NGOSS) framework in Java/J2EE environments.

The OSS/J program has developed specifications for a number of Java API's to easc the integration of Telecoms Operational Support Systems. The API's are collaboratively developed using the Java Community process and released as both Java Interface specifications and supporting XSD definitions. Fully implemented OSS/J interfaces support both Java Value Type (JVT) interactions by means of interfaces defined using an EJB facade pattern and Message Orientated Middleware (MOM) style interfaces using XML messages transported using JMS (Xml Value Type - XVT). Going forwards new API's will be defined with profiles for Web Service (WSDL) style interactions and the Web Services profile will be retrospectively applied to existing interfaces over time (*OSS/J 2006b*).

The earliest and, to date, the most widely deployed of the OSS/J API's are the Quality of Service (Qos) (*OSS/J 2002b*) and Trouble Ticket (TT) (*OSS/J 2002a*) interfaces. Subsequent to the initial definition of these interfaces, OSS/J finalised a Core Business Entities (CBE) (*OSS/J 2006a*), (*OSS/J 2007a*) model in order to provide a more consistent data model which was aligned with the TMForum SID. The newer OSS/J API's closely reference the CBE but the earlier TT and Qos interfaces are not 100% aligned with the current CBE. OSS/J are in the process of specifying an aligned Trouble Ticket interface and separate Fault Management and Performance Management interfaces which will supersede the Qos API. However at the time this project started, the new specifications had not been released and it was preferred to work with an already finalised specification. The new Fault API is substantially similar to the Qos API and future support of the Fault API should be possible without a major re-write of the interface. The design has chosen not to implement Qos interface functionality which is known will not be supported by the new Fault interface. (In particular it has avoided using the XmlSerialiser methods which have been depreciated in later releases of OSS/J).

The OSS/J Quality of Service (Qos) specification defines API's for accessing Performance Management (PM) and Fault Management (FM) data on a Network management System (NMS). It also defines a set of events which the NMS can publish to a Topic using a JMS provider to inform clients of changes in state. The PM and FM sides of the interface can be implemented separately.

The FM interface exposes an alarm list formatted according to the ITU-T X.733 (*ITU-T 1992c*) standard and also generates events as JMS messages corresponding changes in the alarm list (e.g. NewAlarmEvent, ClearedAlarmEvent, AlarmAcknowledgeEvent and

AlarmChangedEvent). Clients of the Qos interface can register with the JMS provider to receive only alarms corresponding to certain filter criteria such as PerceivedSeverity or ManagedObjectType and Instance. This allows for a truly distributed system where clients only register to receive messages relevant to their purpose.

It is possible for Clients to maintain a local copy of the state of an alarm list using events alone which avoids the performance hit and potential latency involved in regularly querying the alarm list. The query interface is required however if the client needs to re-synchronise its view of the alarm list, say when it is starting up for the first time.

F.2.2 Design Philosophy

The qosdaemon project was initiated with a view to making the OpenNMS project more attractive for Telecoms applications where integration with other OSS systems is an important consideration. As a research project, it was also desirable to offer a platform where OSS/J could be demonstrated as providing useful functionality and which would provide a vehicle for the wider community to implement and experiment with NGOSS type solutions in an Open Source environment.

The JCP process used by OSS/J requires that the specifications are released with reference implementations and Technology Compatibility Kits (TCK's) to test implementation's against the specification. Thus the design goals of the OSS/J reference implementations are to provide complete and accurate implementations of the OSS/J specification but not necessarily to provide re-usable libraries or end user functionality. Also, once complete, the OSS/J Reference Implementations are not necessarily subject to ongoing enhancement or maintenance by the original developers.

By way of contrast, the design goals of this project have been to utilise the OSS/J specifications to realise useful functionality in order to allow practical use cases to be demonstrated. Thus the focus has not been on accuracy or completeness of specification implementation but on the realisation of a viable end-user use case using OSS/J functionality within a community sustained open source project. It is very important to realise that two key mantras of successful open source projects are; 'Release early and release often.' and 'Make it easy for potential users or contributors to assess and begin using your offering'.

Therefore the design philosophy of the project has been; Firstly, to choose a use case for OpenNMS which would leverage OSS/J and would deliver immediate value to the OpenNMS community. Secondly, to contribute the solution in such a way as it is fully sustainable as a mainstream contribution to the OpenNMS project even in its initial release. Thirdly, to only implement enough OSS/J functionality as was necessary to support the use case. Finally, to structure the design in such a way as it will be possible for future contributions to go back and complete or address any non-conformances in the design of the interface.

A key aspect of the design has been the separate packaging of libraries (OSSbeans) which could be generally useful for OSS/J implementations from the OpenNMS specific interface code (qosdeamon). The OSSbeans libraries are in a separate Apache 2 licensed project on Sourceforge (*OpenOSS 2007*). The qosdaemon is fully integrated into the OpenNMS code tree which uses Maven to incorporate the OSSbeans dependencies into the OpenNMS build. By this means it will be possible for the development of the OpenNMS and the OSSbeans projects to proceed asynchronously and it will also be possible for other projects to leverage OSSbeans. However the dependence of the OpenNMS Qos interface on the OSSbeans libraries should ensure that there is sufficient community interest to sustain and carry forwards the OSSbeans project regardless of other users.

A key advantage of this design approach has been that the discipline of having to integrate into a real application (OpenNMS) has made OSSbeans more useful as a library. Throughout the process, the design has gone through multiple re-factoring steps in order to partition the functionality so that it can easily be picked up by an external application. A second advantage arises from the fact that designing the initial offering of OSSbeans 2.1.0 has been something of an education. It is possible to see numerous areas where the design could be improved. Thus given sufficient interest, future releases of OSSbeans will be able to learn the lessons of the first implementation in order to provide a more generally useful library. In addition the issues around OSS/J compliance are confined to the OSSbeans library which can have its own roadmap towards compliance as contributions fill in the gaps.

F.2.3 OSS/J Conformance

This project provides partial implementation of the OSS/J QoS interface for OpenNMS. It is offered as an illustrative and training tool to explain OSS/J and to gauge interest from the OpenNMS community in taking the project forwards.

This project leverages the OSSbeans project which provides the core classes for the OSS/J implementation. OSSbeans are a separate project hosted as part of the University of Southampton OpenOSS initiative (*OpenOSS 2007*).

The interface is based upon the OSS/J QoS specification available at www.ossj.org. The basic principles and design patterns of the specification are implemented. However not all of the mandatory functionality is complete and the interface has not been tested against the OSS/J SDK.

Where functionality is provided it does so using classes implementing interfaces conforming to the javax.oss interface tree and the XML messaging uses messages conforming to the OSS/J Qos XSD's. This provides a firm basis for moving towards full OSS/J compliance in future releases.

Some work was done previously by the Budapest University of Business and Technology (BUTE) to demonstrate that the PM interface could be implemented for OpenNMS. This work

has not been incorporated into the present Qos interface but could be taken forwards later. The present Qos interface only implements FM functionality. (Note however that it is still possible for OpenNMS performance threshold crossing events to be converted into OSS/J faults reported by OpenNMS)

F.2.4 Production Use

The interface should be considered experimental and is not optimised for high load environments. Although included with OpenNMS, it can be completely disabled and will not then interfere with other OpenNMS components. However even in its present form the interface may still be useful for some production solutions. Envisaged uses include;

- Integration of OpenNMS alarms with other Operational Support Systems using J2EE or JMS
- Monitoring important alarms from remote OpenNMS systems - potentially on a customer's site. (Note that to circumnavigate firewalls JBossMQ can be configured to send JMS messages using HTTP - although this has not been tested.)

As an example, the interface has been successfully used to integrate OpenNMS with the StateWise Alarm/ Topology correlation engine from Sidonis (*Sidonis 2007*) as part of a proof of concept for managing a Digital TV network

F.2.5 Licence

The qosdaemon project builds an OSS/J interface for OpenNMS. It is released with OpenNMS under the GPL licence and uses code contributed to OpenNMS by the University of Southampton.

OSSbeans are released under the Apache-2 licence by the University of Southampton.

F.3 OSSBeans Project

As describe above, originally the OSS/J and OpenNMS code was developed separately from OpenNMS and contained in a single project on Sourceforge called OpenOSS. To use this code it was necessary to install it as an overlay to existing OpenNMS installations as additional JARs and configuration files. This might have been a satisfactory solution if OpenNMS was reasonably static. However OpenNMS was going though a great deal of churn with its technology refresh program and it was felt that the only way to guarantee the ongoing life the code would be to contribute it to OpenNMS for inclusion in the code base. Thereafter the OpenNMS project could continue to maintain the code and any changes to OpenNMS which affected the OSS/J interface would be picked up through compilation errors.

However this raised a second problem in that the contributors and users of the OSS/J code developed under the OpenOSS project would ideally want to be able to use it apart from OpenNMS and under a more commercially friendly licence than GPL. It was thus decided to split the project into an OSS/J specific library called OSSbeans and an OpenNMS specific contribution which would live within the OpenNMS project.

The need for a library to help implement OSS/J interfaces was highlighted by the early experience of trying to use the OSS/J reference implementations to provide some useful

starting code. Not only were the various OSS/J RIs unfavourably licensed under the JCP, but they also were inconsistently designed using different technologies and did not provide any guidance as to their use. These were not really designed to help implementers of OSS/J build their own interfaces.

OSSBeans sought to provide classes which implemented the management of an OSS/J Qos Alarm List within the Spring framework. The list could be accessed using both the JVT and XVT access methods. Any changes to the Alarm list would result in JMS messages sending notifications to registered clients. Any received messages could modify the Alarm list.

The OSSbeans Library provides mechanisms to marshal and un-marshal OSS/J XML messages into their corresponding javax.oss JVT class implementations. An OSS/J interface specification provides both a java class library and the corresponding XSDs for the XML messages. Unfortunately there is no easy mechanism to convert an OSS/J XML message into its corresponding javax.oss class and vice versa. XML Marshalling frameworks such as XMLBeans (*XMLBeans 2006*) and Castor (*Castor 2007*) are available but these convert the XML into a framework specific set of classes which do not inherit from or implement JVT classes corresponding to the OSS/J interface. Thus it was necessary to build a marshalling system which generated classes from the XSD.

It took considerable experimentation with several open source marshalling frameworks before it was possible to work out how this was to be done. The examples given with the marshalling frameworks did not cover anything as complex as the OSS/J interfaces. The XML marshalling and un-marshalling for the interface was eventually developed using XMLBeans (*XMLBeans 2006*). The XMLBeans framework was originally developed by BEA and contributed as open source to the Apache foundation. This provided a rapid code generation framework which could generate raw java types based upon the OSS/J schema documents.

The marshalling system required the design of a three phase pre-compilation processing system using XMLBeans to generate all the necessary classes from the supplied XSD specifications before the main compilation phase (Figure 38). This gradually builds up the dependant classes required by the increasingly complex data types. Compiling the library directly from the specification XSD's meant that the XML messages generated were guaranteed to be correct according to the specification. Additional classes were then written to help with marshalling the more complex java objects representing alarm lists or query messages.

The class diagram of the library is given in Figure 39. The basic component of the library is an OSSbean which is a class implementing the Runnable interface and the JMX Mbean interface. An OSSBean can thus be started as a separate thread managed by JMX and can also be run in OpenNMS as a separate daemon. The OSSBeanRunner class can start and stop a collection of OSSBeans in separate threads according to the OssBeanRunnerList which is a list of OssBeans to run given to an OSSBeanRunner by a Spring ApplicationContext.

An OSSBeanTransmitter is a class which registers with a JMS provider to send OSS/J messages. An OssBeanReceiver is a class which registers with a JMS provider to receive OSS/J messages. When it receives an OSS/J message it passes the message to an appropriate AlarmEventRecieverEventHandler which is implemented by the user of the library.

When used in OpenNMS, the qosdrx daemon sets up multiple OssBeanReceivers each listening to a different JMS topic. The qosdrx implements an AlarmEventRecieverEventHandler which modifies the OpenNMS Alarm List on receipt of a message. Similarly, the qosd extends the OSSBeanTransmitter class in order to transmit a message whenever the OpenNMS alarm list changes.

This section has provided a simple introduction to the OSSBeans library. Further details on how this works are provided in the Javadoc and other documentation included with the qosdaemon contribution to OpenNMS.

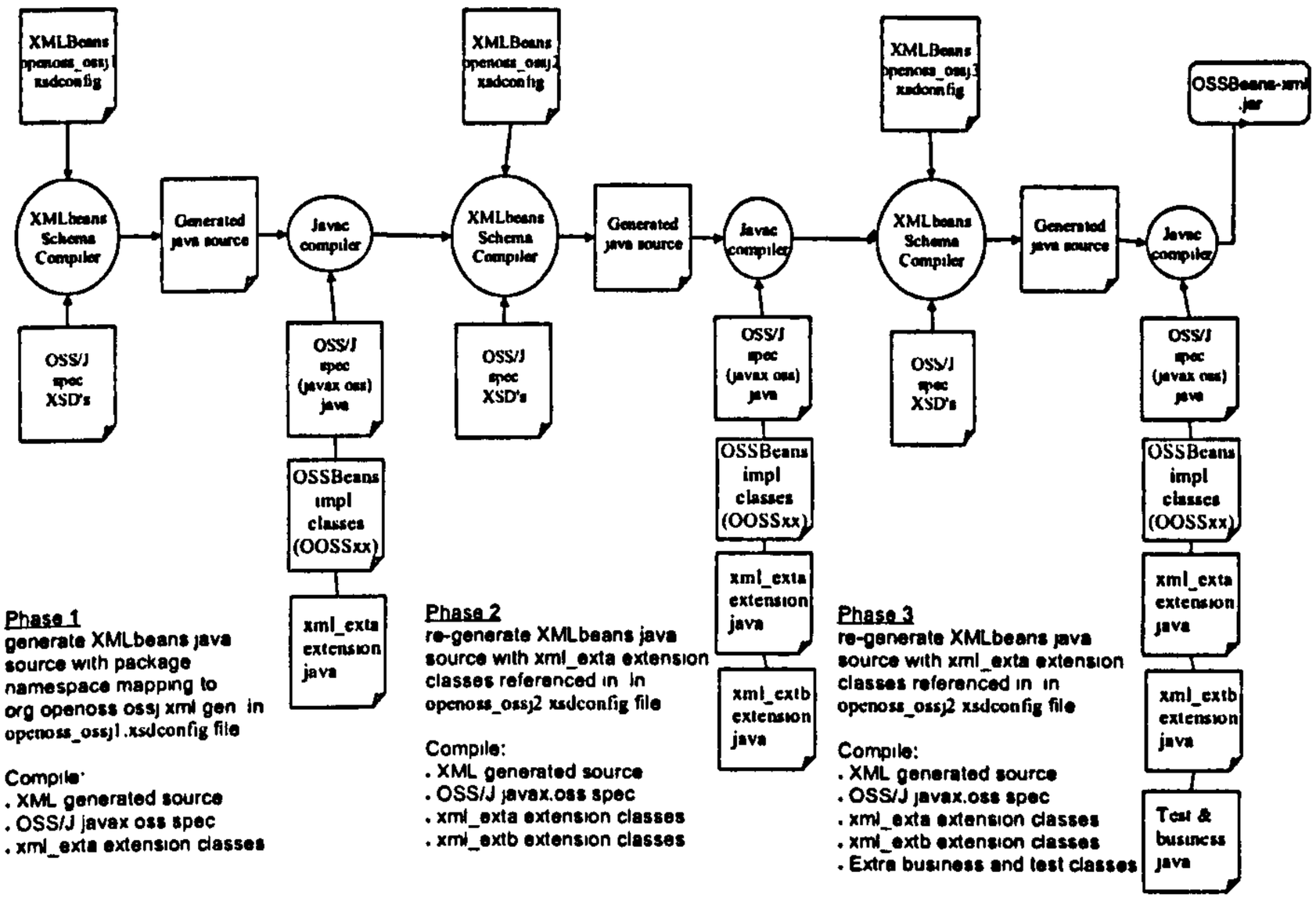


Figure 38 OSSBeans - compiling a marshalling framework from OSS/J XSD's

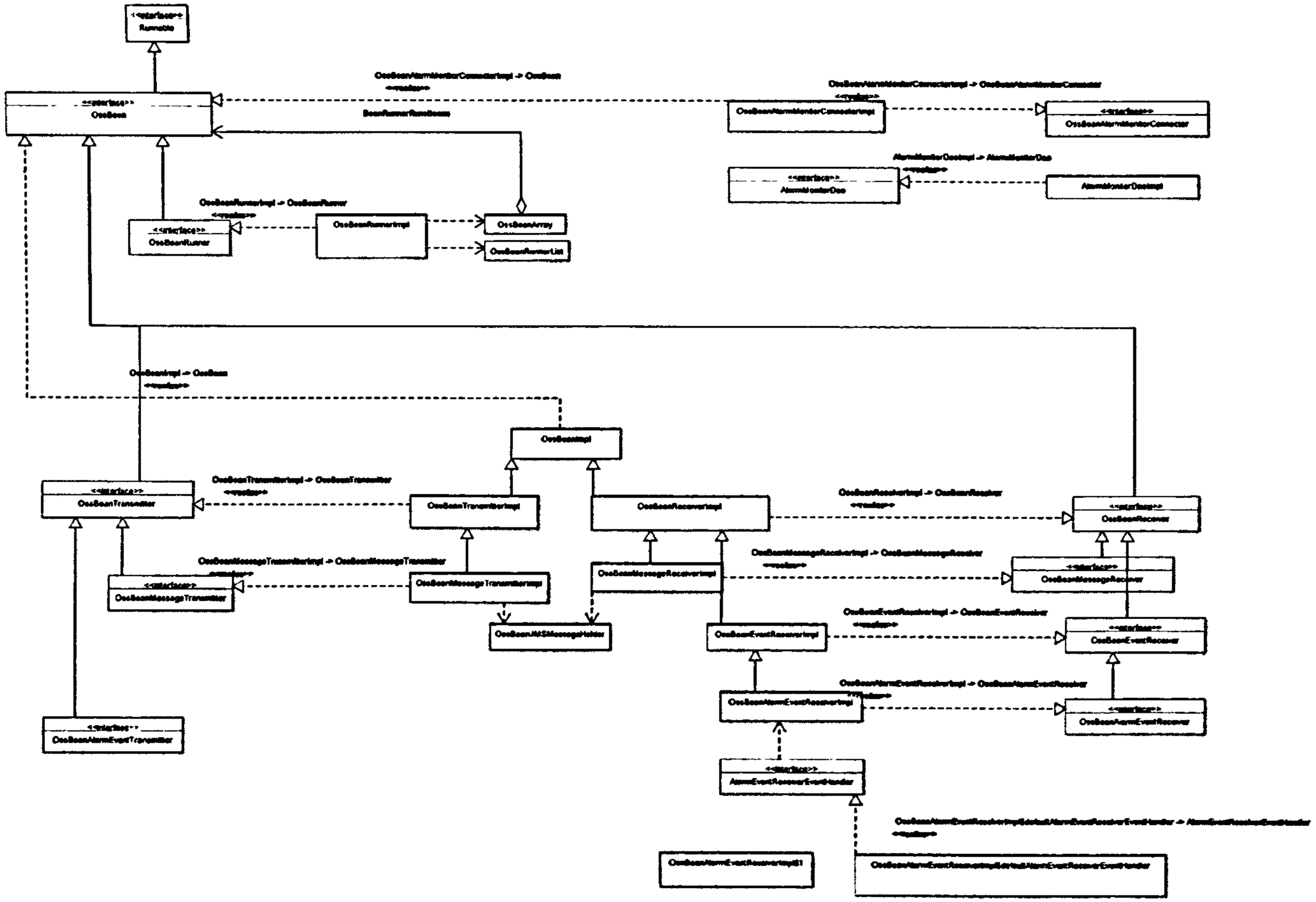


Figure 39 OSSBeans class diagram

F.4 OpenNMS qosdaemon

How the qosdaemon is used can be further understood by referring to its use in the DVB-T proof of concept in Chapter 5.

F.4.1 Functionality Provided

The current release of the qosdaemon module uses OSSbeans Release 2.1.0 and provides two OpenNMS daemons which can be used independently or together. The daemons share a data access layer, OssDao which is a data access object which maps the OSS/J Qos interface onto OpenNMS's internal alarm list exposed by the OnmsAlarmDao.

The qosd daemon publishes the internal OpenNMS alarm list as an OSS/J alarm list. The qosdrx daemon allows an OpenNMS system to connect using the OSS/J interface to remote OpenNMS systems running qosd. This allows a 'master' OpenNMS to monitor the state of the alarms lists in 'slave' OpenNMS systems. The present implementation is almost exclusively JMS event driven with limited alarm list query functionality provided as a J2EE JVT option on qosd.

The implementation leverages JBossMQ as the JMS provider. In theory other JMS providers could be used but these have not been tested.

F.4.2 QoSD daemon

The QoSD daemon monitors the OpenNMS alarm list and generates OSS/J JMS events corresponding to changes in the state of the alarms in the list. It can run in two modes; natively on OpenNMS (Appendix F.4.2.1) or in conjunction with a separate J2EE OSS/J JVT component (ear) running in an application server (Appendix F.4.2.2).

F.4.2.1 Native OpenNMS provided interface.

OpenNMS does not run natively in a J2EE container but leverages the Spring framework and JMX to provide a container-like environment for its daemons. The qosd daemon code can run natively as a Spring application within OpenNMS as illustrated in Figure 40. In this case it uses the OSS/J XVT (XML over JMS) profile to publish alarm list changes. The qosd daemon publishes OSS/J AlarmEvents as both JMS TextMessages (XML) and as JMS ObjectMessages (JVT) containing AlarmEvent objects.

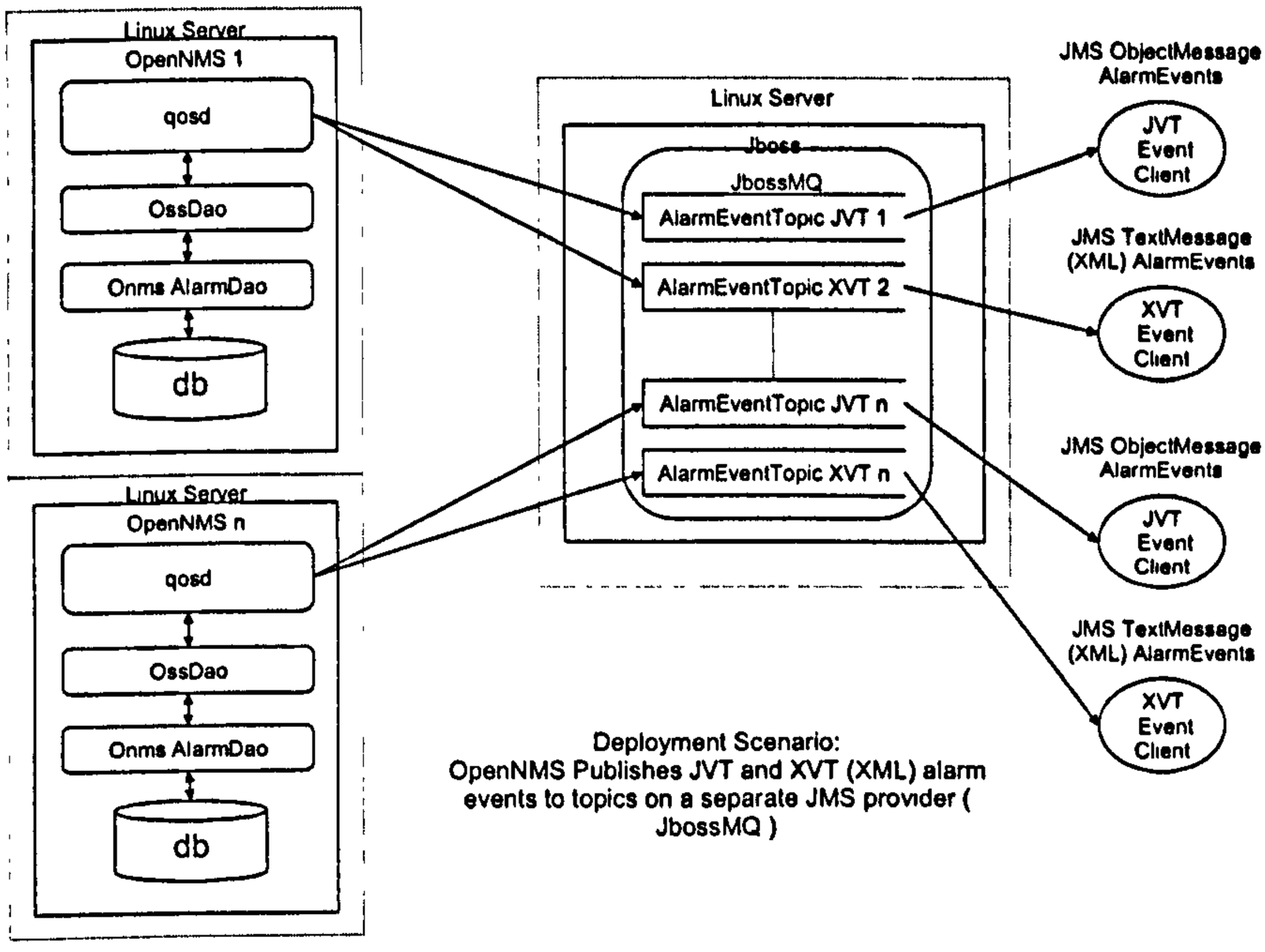


Figure 40 Native OpenNMS provided interface.

F.4.2.2 Separate J2EE server provided interface.

An alternative configuration is possible where the qosd daemon connects to an EJB application running in a separate J2EE server as illustrated in Figure 41. This application is known as OSSBeans-qos-ear and is available from the OSSbeans site. In this mode the EJB exposes OSS/J semantics and allows external applications to connect with the EJB as an OSS/J JVT interface. Note that only very limited alarm list query functionality is currently provided (query for all alarms).

Note that this configuration requires a J2EE server (JBoss) to be hosting an OSSBeans-qos-ear instance locally to each OpenNMS implementation which is running qosd in this mode. In most circumstances, it is easier to use the native XVT interface discussed in Appendix F.4.2.1 for the remote machines as they can all use a single JBossMQ deployment and a local J2EE server is not required for each remote OpenNMS.

Which mode qosd is running is determined by a setting in the opennms.conf file:

- To use the native OpenNMS provided interface use -Dqosd.usej2ee=false
- To use the separate J2EE server provided interface use -Dqosd.usej2ee=true

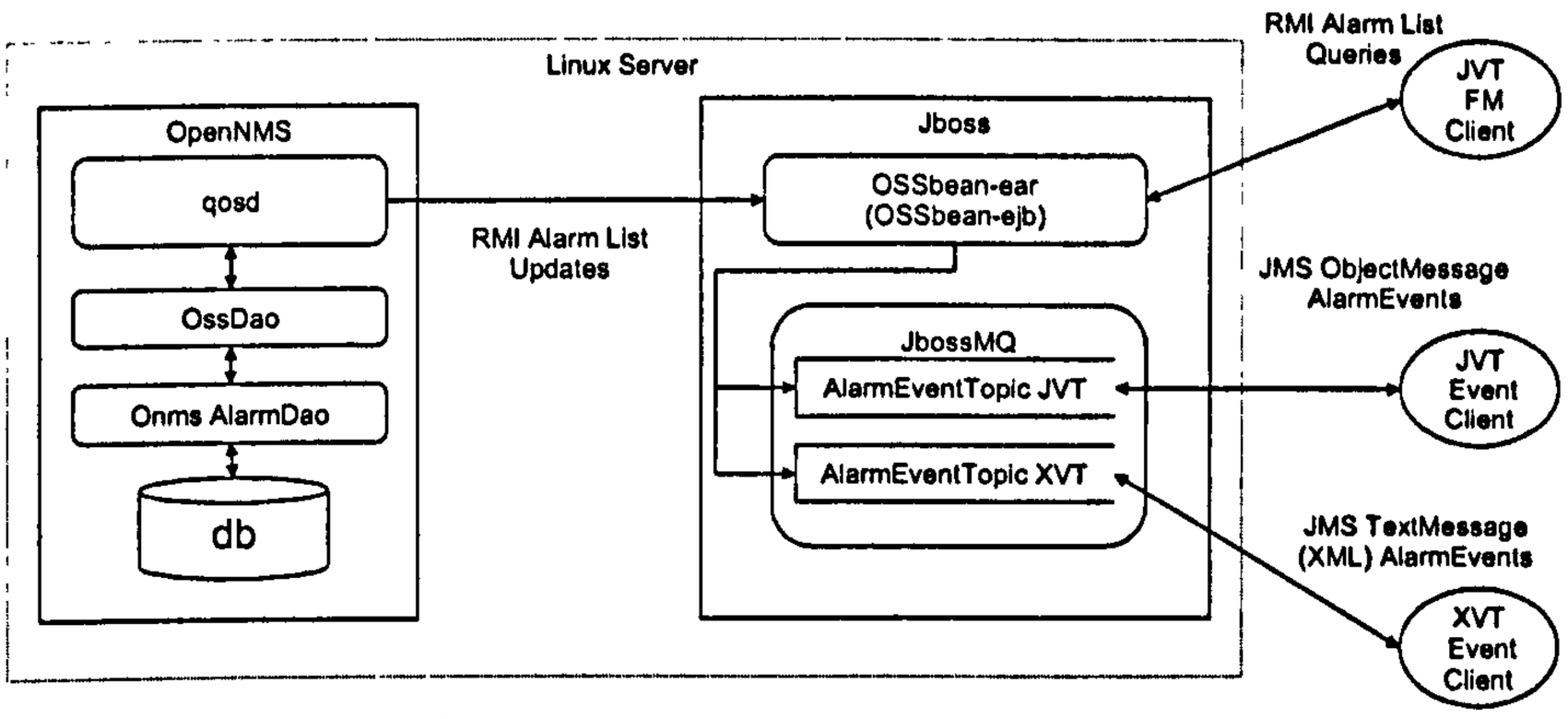


Figure 41 Separate J2EE server provided interface.

F.4.3 QoSDrx

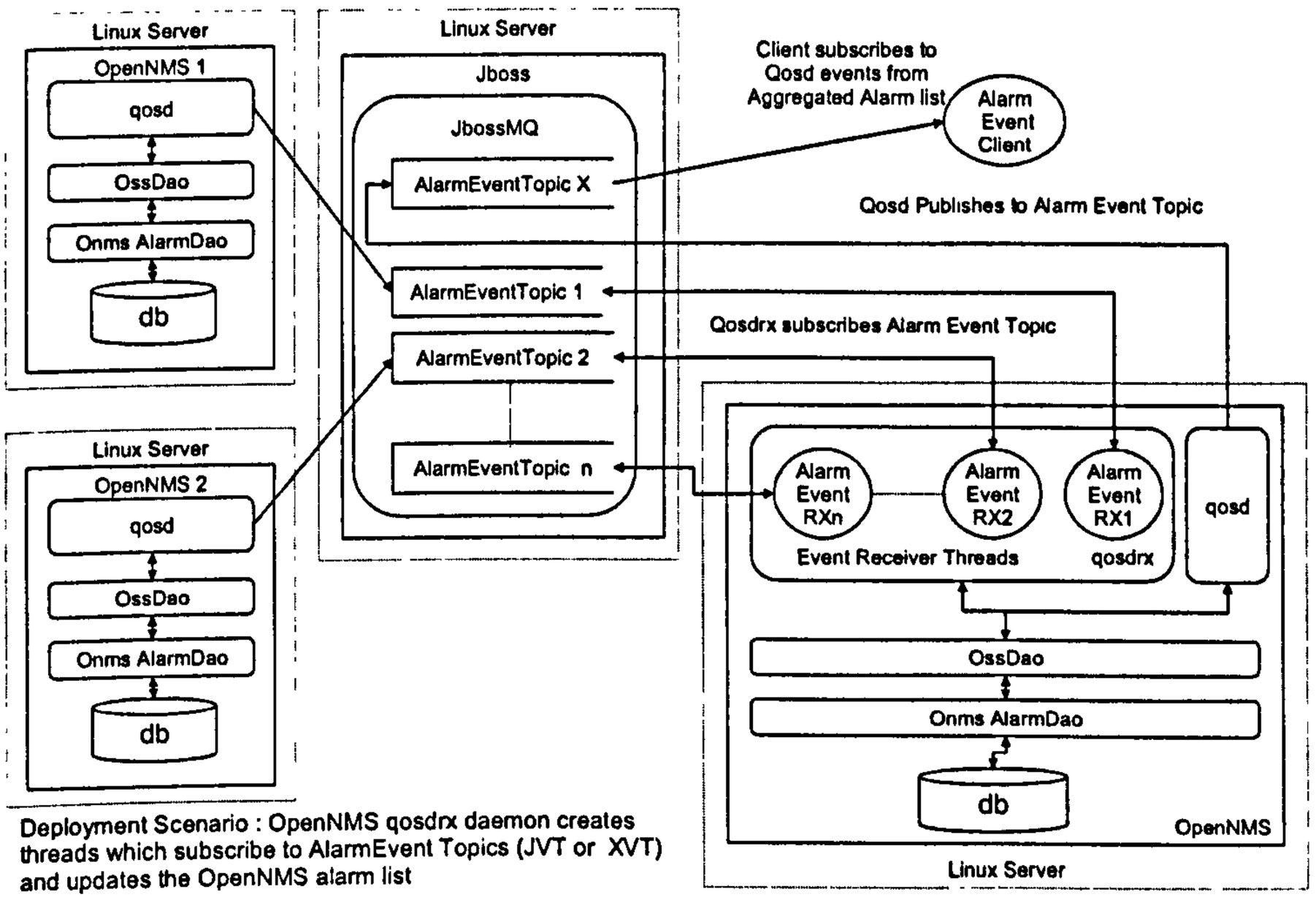


Figure 42 Illustrating remote slave OpenNMS systems talking to a central master system

The qosdrx daemon can connect to multiple OSS/J event topics hosted on a JBossMQ server and receive OSS/J alarm events from remote OpenNMS systems running qosd. This is illustrated in Figure 42. The local alarm list will be updated to reflect the remote alarm lists. Note that no resynchronization capability is provided at this time so it is possible for the alarm lists to get out of alignment if messages are lost. However in practice, the JMS messaging system should provide a reliable transport.

F.5 OpenNMS qosdaemon Package Description

This section provides additional information on the qosdaemon which implements the OpenNMS OSS/J Qos Interface.

F.5.1 Functionality Overview

The qosdaemon provides an elementary OSS/J Qos interface for OpenNMS.

For more information on the OSS/J Qos Specification please see the OSS/J site www.ossj.org.

For more details on configuration and functionality provided please see the documentation and example configuration in the OpenNMS directory.

$opennms_home/contrib/qosdaemon

The interface can run natively in OpenNMS and use OSS/J XML messages to reflect the state of the OpenNMS internal alarm list or it can use an external EJB running in JBoss to expose the alarm list as JVT objects. Currently only OSS/J XVT and XML AlarmEvents have been implemented. A future release will service OSS/J XML queries. The interface uses the services of an external JMS provider. Although the JBossMQ implementation of JMS is used, it should be fairly easy to substitute another JMS implementation.

This interface is in two parts which can be run separately as two OpenNMS daemons;

qosd is an OSS/J server which exposes the OpenNMS alarm list as an OSS/J alarm list to other systems. The changes to the local alarm list are exposed as JMS XML messages which can be used to update an alarm list in a remote client.

qosdrx is an OSS/J event client which can listen for alarm update events from other servers. It updates the local OpenNMS alarm list based upon the received events.

During the development of the qosdaemon, OpenNMS migrated to using the Spring framework. Spring has been adopted extensively in the design of the qosdrx daemon. However some of the code in the qosd daemon pre-dated the introduction of Spring and has been adapted to use Spring rather than re-written. This has lead to a somewhat convoluted initialization process for qosd which will be tidied up in a later release. In general most classes use dependency injection and Spring wiring to get their dependencies and most classes are exposed as implementations of interfaces with the 'impl' suffix to their name. Where classes have been re-implemented or gone through several versions, the implxx suffix is used where xx indicates the version. This has allowed several versions of the same class to co exist and be wired in for testing etc during development. (As an example see QosDimpl2.java)

F.5.2 Package Overview

F.5.2.1 OssDao

Overview

The OssDao package provides a buffer between the internals of OpenNMS and the requirements of the OSS/J code. The internal alarm definition in OpenNMS contains different fields and severity values to the OSS/J X.733 definitions. This mapping is (mostly) confined to the OnmsAlarmOssjMapper class.

The OpenNMS alarm definition has been extended to allow it to represent alarms from remote systems. Each OpenNMS alarm now contains an ApplicationDN which identifies the unique name of the remote system which generated the alarm and an OssPrimaryKey which identifies the local identifier for that alarm on the remote machine. When an alarm update event is received, the OssDao searches for an existing alarm with matching ApplicationDN and OssPrimaryKey in order to do the update. (If ApplicationDN is blank then the alarm is assumed to be from the local machine).

Each OSS/J alarm contains an identifier for the managed object which issued the alarm. This is always mapped to an OpenNMS node. The Asset Register entry for nodes now has extra fields to uniquely identify the managed object system wide (ManagedObjectInstance and ManagedObjectClass). The OssDao provides methods to do a rapid reverse look up on the Asset register to find nodes with a unique id which is given by the concatenation of the ManagedObjectInstance and ManagedObjectClass.

Build Directory / package	Class / XML configuration	Description
org.openoss.opennms.spring.dao	OnmsAlarmOssjMapper	This class provides methods to map OSS/J AlarmValue to OpenNMS alarms. If the OpenNMS alarm model changes, then most of the changes to the qosdaemon will happen here
	OssDaoOpenNMSImpl	This class provides a single point of access to the OpenNMS alarm list and node asset table for both the Qosd and Qosdrx. It provides a cache which allows for rapid look up of alarms by ApplicationDN and OssPrimaryKey and of nodes by managedObjectInstance and managedObjectType
	OssDaoOpenNMSImplSingleton	This class provides a wrapper for OssDaoOpenNMSImpl which allows it to be created and initialized as a singleton by either qosd or qosdrx depending on which daemon starts first. Once initialized both daemons can use the same instance of the DAO.

Note that when the OssDao is initialized, the QoSDrx log file will record any nodes having no data or duplicate data in the ManagedObjectInstance and ManagedObjectType fields of the Asset table. If the ManagedObjectInstance and ManagedObjectType fields of nodes are empty or duplicated the OSS/J Alarm management will not work correctly.

F.5.2.2 QoSD Daemon

Overview

The QoSD Daemon realises an OSS/J QoS server interface by mediating between the OpenNMS internal alarm management model and an OSS/J implementation realized using the OSSBeans OSS/J Qos library. The OSSbeans AlarmMonitor bean can run natively in the OpenNMS Spring container or in a remote the JBoss J2EE server. If it is running in the remote J2EE server then it exposes its alarm list according to OSS/J semantics as JVT objects through a facade bean. Note that presently only a very simple query is supported to get the entire alarm

list. The AlarmMonitor bean also sends out XVT (XML) and JVT AlarmEvents as the alarm list changes. In the future it is intended to extend the OSSBeans alarm monitor to also support full JVT and XML queries. If the AlarmMonitor bean is running natively in OpenNMS, it cannot expose the OSS/J JVT interface and only generates the JMS events however it is envisaged that this is the way most users will want to use the interface.

How it works

The qosd.properties file contains the address information for the QoSD to connect to its JMS queues. The configuration for the QoSD daemon is picked up from the /etc/qosd.properties file using the singleton PropertiesLoader class. (In the future this could be picked up using the Spring application context in a similar manner to the qosdrx code.)

The QosD Daemon registers with the OpenNMS eventd daemon to receive events using the call;

```
eventIpcManager.addEventListener(this, ueiList);
```

The events which qosd responds to are determined by the ueiList object which is configured by the QosDConfigFactory using and the QoSDConfiguration.xml file in the OpenNMS /etc directory. Castor is used to un-marshal the XML into the ueiList according to the QoSDConfiguration.xsd file.

When an event which qosd has registered for occurs its onEvent() method is called. Qosd responds to two types of events. Firstly any events corresponding to changes in the asset register are used to force an update in the OssDao node cache. Secondly an 'alarm changed event' causes the qosd to review the local OpenNMS alarm list for changes.

The qosd depends upon the vacuumd daemon which must be configured with an automation to look for new or changed alarms in the local alarm list and throw an 'alarm changed event' if a change is detected. An example vacuumd-configuration.xml file is included in /contrib/qosdaemon/qos-example-configuration.

The vacuumd automation also implements the X733 alarm life cycle whereby OpenNMS correlates alarm raising and alarm clearing SNMP traps and removes cleared and acknowledged alarms from the alarm list.

Qosd responds to an alarm changed event by causing the OssDao to update its cache of OnmsAlarm values with a new snapshot of the alarm list in the database, translating the OnmsAlarm values to OSS/J AlarmValues using the OnmsAlarmOssjMapper and then forwards this new list of alarms to the OSSBeans AlarmMonitor bean. The AlarmMonitor bean compares this new list with its current list and sends out AlarmEvents when alarms in its current list are added, deleted or changed to match the new list.

The AlarmListConnectionManager interface determines whether the remote J2EE or local Spring implementation of the AlarmMonitor is used. Two implementations of the AlarmListConnectionManager are available to be wired in using Spring;

AlarmListConnectionManagerJ2eeImpl.java connects to the remote bean using RMI.
AlarmListConnectionManagerSpringImpl.java implements the bean locally.

This choice of which connection manager to use is determined by the setting of the system variable -Dqosd.usej2ee=[true]|[false] in the /etc/opennms.conf file which is read when the qosd is initialized.

In a heavy alarm storm, vacuumd could trigger multiple alarm changed events such that qosd initiates multiple parallel look ups to the database which would be very costly. The OpenNMSEventHandlerThread is used to avoid this eventuality by latching any new 'alarm changed events' while a database look up is being performed. Thus a lookup has to complete before a new one is initiated. All of the changes which occur during this time will be picked up on the next alarm retrieval.

Build Directory / package	Class / XML configuration	Description
org.openoss.opennms.spring.qosd:	AlarmListConnectionManager	Interface defining how qosd interfaces with OSSBeans AlarmManager bean
org.openoss.opennms.spring.qosd.ejb	AlarmListConnectionManagerJ2eeImpl AlarmListJ2eeConnectionManagerThread	Implementation of interface to remote AlarmMonitor bean on JBoss. ConnectionManagerThread establishes RMI connection to J2EE server and waits for server to be available if connection is lost
org.openoss.opennms.spring.qosd.spring	AlarmListConnectionManagerSpringImpl	Implementation of interface to AlarmMonitor bean running locally in OpenNMS
	QoSD.java QoSDimpl2.java	Interface and implementation of QoSD daemon.
	OpenNMSEventHandlerThread	Provides a latch for events which occur while a database lookup is happening
	QoSDConfigFactory.java UEIHandler.java PropertiesLoader.java	Classes for loading the configuration of QoSD
./src/main/castor	castorbuilder.properties QoSDConfiguration.xsd	Configuration for Castor XML marshalling framework. Defines the format and contents of the QoSD-configuration.xml file which sets up which OpenNMS events the qosd daemon listens for
org.openoss.opennms.spring.qosd.jmx:	QoSD.java QoSDMBean.java	All OpenNMS daemons are started as JMX beans in the same way. The QoSD.java and QoSDMBean.java in the qosd.jmx package are used by OpenNMS to initiate new threads for running the daemon and to pass in its Spring application context in order to allow access to OpenNMS DAOs and other daemons.
./src/main/resources/org/openoss/opennms/spring/qosd	OssjTypeSpecificationApplicationContext.xml	This Spring application context file is used as a factory to configure the default settings for the OSS/J AlarmValues before they are populated with alarms.

Build Directory / package	Class / XML configuration	Description
	qosd-j2ee-context.xml qosd-spring-context.xml	These application context files are used to either set up a local AlarmMonitor bean or a remote AlarmMonitor bean on a J2EE server. Which application context is used is determined by the setting of the system variable -Dqosd.usej2ee=[true]|[false] in the /etc/opennms.conf file
org.opennms.web.alarm	Alarm	A copy of the org.opennms.web.alarm.Alarm class is included in this project to provide a severity mapping for OpenNMS alarms. This is done because at the time of writing OpenNMS did not generate a webapp JAR which could be accessed through MAVEN. If the webapp is made into a Maven package then this can be replaced with a reference in the pom file. Alternatively (and better) OpenNMS could include the severity mapping centrally in the OnmsAlarm model.

F.5.2.3 QoSDrx Deamon

Overview

The QoSDrx daemon implements multiple OSS/J JMS event listener clients for OpenNMS. Each client connects to a different OSS/J JMS AlarmEvent Topic on a JMS provider (tested with JBossMQ but other providers could be used). Alarm Events are used to update the local OpenNMS alarm list such that it mirrors the alarm event list in the remote OSS/J servers. The clients are fully configurable using an XML configuration file in the /etc directory. Planned future extensions to the Daemon will allow it to also synchronize with remote servers using OSS/J XML queries.

How It Works

The QoSDrx Daemon realises an OSS/J QoS client interface using the OSSBeans OSS/J Qos library. The Library provides a collection of classes for implementing clients and servers for receiving or transmitting OSS/J messages as XML or JVT objects. Fundamental to the library is the concept of an OSSbean which is Spring class which can be run as a separate thread for handling OSS/J messages and is configured using a Spring application context XML file.

As with the QoSD daemon, the qosdrx.jmx classes (QoSDrx and QoSDrxMBean) are used by OpenNMS to spawn the qosdrx daemon and initialise an OssDao if it does not already exist. The daemon passes the OpenNMS application context to the OSSBeans OSSBeanRunner class which reads the /etc/QoSDrxOssBeanRunnerSpringContext.xml file to generate an application context and spawn separate threads for each defined OssBeanAlarmEventReceiver class. Each OssBeanAlarmEventReceiver then registers to listen to its defined JMS AlarmEventTopic and waits to receive OSS/J AlarmEvents (both XVT and JVT events are supported). If any receiver cannot attach or gets disconnected from its JMS Topic, it waits for a defined timeout and then tries to reconnect.

The OssBeanAlarmEventReceivers need to have AlarmEventReceiverEventHandler classes assigned which are used to perform appropriate actions on reception of each event. The QoSDrx daemon defines an AlarmEventReceiverEventHandler implementation which interacts with the OssDao to update the OpenNMS Alarm list database on each received event. (The present QoSDrx demon AlarmEventReceiverEventHandler implementation is called QoSDrxAlarmEventReceiverEventHandlerImpl2)

The intention of this design is to extend the OSSbean concept in the future to make each OSSbean also a JMX Mbean. This would allow finer real time monitoring and management of each receiver client. However in the present design all of the OSSbeans are spawned off the root OpenNMS JMX bean which is able to report the statistics from all of the beans using the OpenNMS MX4J console.

Alarm Life cycle

Note that the present use case only utilizes NewAlarmEvent and AlarmCleared events to update the Alarm list. Other AlarmEvents are simply logged. However the QoSDrxAlarmEventReceiverEventHandlerImpl can easily be extended to use more of the OSS/J alarm event types if a tighter coupling between client and server is required. The following process is followed by QoSDrx to process AlarmEvents;

1. When a NewAlarmEvent message is received, its ApplicationDN and OssPrimaryKey are checked against the OpenNMS alarm list. If the alarm exists, there is an error which is logged and the NewAlarmEvent message is ignored (the NewAlarmEvent messages should refer only to a new not an existing alarm).
2. If the alarm is new (i.e. not in the AlarmList), QoSDrx has two modes of operation based upon how it is desired to handle the managedObjectInstance and ManagedObjectType fields in the incoming messages.
    a) Alarm can be logged against a node which is named to represent the AlarmTopic. Each OssBeanAlarmEventReceiver is given a unique name in the QoSDrxOssBeanRunnerSpringContext.xml file. In this case OpenNMS should be configured with nodes named after the name of each OssBeanAlarmEventReceiver. When an alarm comes in to a given topic, the alarm is logged in OpenNMS against a node named with the same name as the OssBeanAlarmEventReceiver. This is a good configuration if you simply want to look at alarms from subordinate OSS/J servers on a per server basis but you don't want to put every managed object in the OpenNMS database corresponding to the managedObjectInstance and ManagedObjectType of all the objects being monitored remotely.
    b) Alarms can be logged against nodes having the same managedObjectInstance and ManagedObjectType fields as the incoming message. In this case the listening OpenNMS must have a database populated with all of the nodes (with the same Asset data for ManagedObjectInstance and ManagedObjectType) as populated in the subordinate OpenNMS's
    Note that in both the above cases, regardless of how the alarm is displayed locally, it will be forwarded through QoSD with the same managedObjectInstance and ManagedObjectType as the original message. (i.e. OpenNMS will not modify the origin of the message even though it displays it against a different node locally)
3. When an AlarmClearedEvent message is received its ApplicationDN and OssPrimaryKey are checked against the OpenNMS alarm list. If the alarm does not exist, there is an error which is logged and the ClearedAlarmEvent message is ignored (the ClearedAlarmEvent messages should refer to an existing alarm).
4. If the corresponding alarm exists in the Alarm List, it will be set to cleared and acknowledged. The vacummd process will subsequently remove the cleared and acknowledged alarm from the alarm list.
    Note that the behaviour of the interface could easily be extended to allow usage of separate acknowledgment messages of cleared alarms before removing them from the list. However this has not been done as it appeared to complicate the use case without much extra value. In the future we will make alternative alarm life cycle behaviour's a configurable item.

Build Directory / package	Class / XML configuration	Description
org.openoss.opennms.spring.qosdrx.jmx	QoSDrx.java QoSDrxMBean.java	As with Qosd used by OpenNMS to start the Daemon.
org.openoss.opennms.spring.qosdrx	QoSDrx.java	Class which initialises the OssDao and loads the OSSbeanRunner in order to start the OSS/J AlarmEvent clients.

Build Directory / package	Class / XML configuration	Description
	QoSDrxAlarmEventReceiverEventHandlerImpl2 QoSDrxAlarmEventReceiverEventHandlerImplShell	QoSDrxAlarmEventReceiverEventHandlerImpl2 provides a handler for OSS/J events received which updates the OnmsAlarm list through the OssDao. Note that an unused QoSDrxAlarmEventReceiverEventHandlerImplShell simple logs all events received. This provides a template for writing any other AlarmEvent handling behaviour.
./src/main/resources/org/opennms/spring/qosdrx	qosdrx-spring-context.xml	Initial Spring context used to start the OSSbeanRunner which then loads the QoSDrxOssBeanRunnerSpringContext.xml file and launches the AlarmEventReceiver clients.

Appendix G - Open OSS Catalyst Final Technical Report

Appendix G OpenOSS Catalyst Final Technical Report

This appendix reproduces the final technical report from the OpenOSS catalyst. It was edited by Craig Gallen based upon input from the team participants and submitted as the final report to the TM Forum.



The 1st step in an alternative approach to drive the implementation of NGOSS

Technical Report

Version 0.3 *July 2005*

©University of Southampton 2005
©TeleManagement Forum 2005

G.1 Executive Overview

This document reports on the work done for the OpenOSS proof of concept demonstrated at Telemanagement world Nice 2005.

The OpenOSS Catalyst was initiated to investigate the viability of an open source research program to trial an implementation of an NGOSS architecture. The intent was not to complete an architecture but to do enough preliminary work to understand the value of such an undertaking and the likely issues which would need to be addressed if such a program went forwards. The business requirements and deliverables for this project were defined in the Opens OSS project charter [1]

The project sponsors and participants were;

Agilent Technologies	Covad
AutoMagic	Invocom
British Telecom	NTT Group
Budapest University of Technology and Economics	QinetiQ
Cable & wireless	Sun Microsystems
Cognisant	University of Southampton
COLT	Vodafone

The project sponsors desired to kick start an investigation into the potential benefits of Open Source when applied to telecoms Operational Support Systems. The intention was not to create a competitive solution to existing OSS products but to investigate whether an open source approach to collaboration could be promoted to reduce the repeating costs of integration between OSS systems.

This document provides an overview of the technical approach taken through the project and some of the key learning's and recommendations from the project.

Notice

This document is released under the Apache licence
Copyright 2005 University of Southampton School of Electronics and Computer Science
Licensed under the Apache License, Version 2.0 (the "License"); You may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

G.2 Introduction

The OpenOSS Catalyst was initiated to investigate the viability of an open source research program to trial an implementation of an NGOSS architecture. The intent was not to complete an architecture but to do enough preliminary work to understand the value of such an undertaking and the likely issues which would need to be addressed if such a program went forwards. The business requirements and deliverables for this project were defined in the Opens OSS project charter [1] (*TMForum 2004c*).

This document provides an overview of the technical approach taken through the project and the key learning's and recommendations from the project.

G.3 Project Objectives

The project sponsors desired to kick start an investigation into the potential benefits of Open Source when applied to telecoms Operational Support Systems. The intention was not to create a competitive solution to existing OSS products but to investigate whether an open source approach to collaboration could be promoted to reduce the repeating costs of integration between OSS systems.

A Telemanagement Forum Catalyst project was chosen as the vehicle for this initial investigation for several reasons. TMForum Catalysts are an established industry mechanism for short term collaborations within a well understood legal and IPR framework. Commercial participants are attracted to catalysts because of the marketing opportunity and the opening to work closely with potential service provider customers. Thus a catalyst provided a means to invite the involvement of players from across the industry and also attract feedback from the results of the work.

Furthermore, a catalyst represents in microcosm the project structure one would envisage for a longer term project and thus allowed us to understand the organisational issues which a more substantial open source project would have to address. How should we create an environment for collaborative research involving industry, the Telemanagement Forum and Academia? What sort of open source licensing would be required? What collaborative tools and processes should be adopted? How should an open source project be governed and in what sense would Catalyst project sponsors represent a governance model.

In a real sense, many of these questions have been addressed elsewhere in other open source projects (see for instance Collaboration, Conflict and Control: The 4th Workshop on Open Source Software Engineering http://opensource.ucc.ie/icse2004/). However, having not previously had an open source strategy, the OSS industry has not in practice really thought about these issues and we hoped that the catalyst would begin to break up the ground.

From a technical perspective we wanted to understand the extent to which an OSS research project could be realised by leveraging existing open source projects along with contributed COTS products. To this end we wished to investigate;

- Were there viable open source network service components with which to create a representative managed network?
- How could we effectively use open source development tools and techniques e.g. Sourceforge CVS, Netbeans, Eclipse?
- Were there viable open source network management components with which to build a representative management architecture (e.g. OpenNMS, Drools , JBPM, JBoss)
- How easy would it be to implement OSS/J interfaces by reusing some of the contributed code within the OSS/J organisation?
- What value would a COTS vendor derive from participating in such a project?

Clearly different potential participants would have different motivations and we wished to understand whether a single project could encompass enough of their aspirations to be viable. We wanted to understand the issues concerning the trade off between the different focuses of groups participating in such a project; research activities, standardisation and reference implementation activities, product development activities.

To what extent would an industry open source project need very clear pre-defined design patterns to be successful or would there be there room for contributors to come up with their own approach within a broad framework. Within the research context, should we allow multiple design approaches to develop in parallel and converge later onto a common design pattern for our work or should we fix the design early on? This would reflect upon the extent to which we used tightly coupled integration against a pre-fixed architecture or loosely coupled integration – encouraging each team to pursue their own designs and offer them into a broad ecosystem of NGOSS or OSS/J implementations.

Finally, we also wished to understand the value which a sister Catalyst – NGOSS/MDA could derive from modelling or work. Could this be an example of creating a body of work which could be built upon by future catalysts?

G.4 Collaborative Environment

As part of our investigation into collaborative working, we wanted to try using tools which were commonly used to help collaboration in the open source community.

From the start we wanted to encourage the use of an electronic bulletin board for constructing, reviewing and sharing project documentation among the team. We chose an open source wiki to fulfil this function based upon software from www.twiki.org. This was hosted at the University of Southampton and was accessible to anyone working on the project.

Using the Twiki methodology for collaboration and documentation (Propose, Discuss, agree and Re-factor) was encouraged across the team and the Twiki was positioned as the most up to date source of information on the project. We subsequently used the artefacts and documentation contributed though the Twiki to construct the final project documentation deliverables (e.g. this document). This allowed the project to develop a shared understanding and common documentation as it proceeded rather than exchanging a storm of emails or multiple marked up documents.

In addition we also agreed to create a project on Sourceforge for storing our most up to date code base as the project proceeded. Again, we wanted to learn about using a tool very commonly used within the open source community.

G.5 Test Network

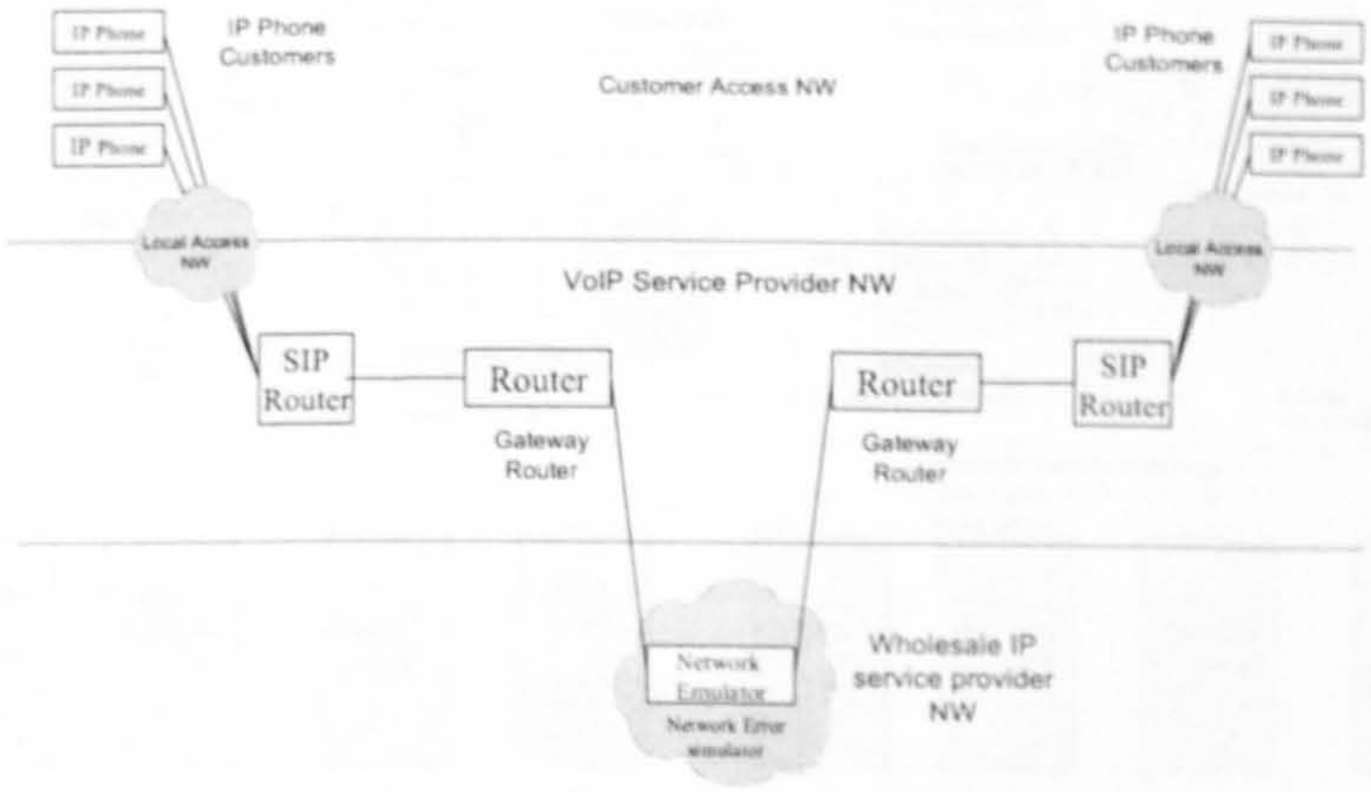
A key objective of the Catalyst was to see if a realistic test service could be constructed using open source components in order to exercise the management solution. In addition to building the service network, we also had to construct mechanisms for introducing known errors into the network in a controlled and repeatable fashion. In this section we discuss the test network design.

G.5.1 Network Scenario

Figure 43 illustrates the simple Voice over IP service provider scenario we targeted. We envisaged that a VoIP service provider would have a number of geographically dispersed nodes interconnected through an outsourced wholesale IP service. Each node would have a SIP router or gateway and would be connected either directly to customer clients or through an outsourced access network.

The VoIP service provider would have their own OSS stack, monitoring the state of the traffic within their network and monitoring the VoIP calls by analysing the SIP signalling between the SIP gateways. The service provider would not have direct access to the status of the wholesale network but would be able to tell if the wholesale network was experiencing problems by monitoring the traffic at the edge of their own network.

To keep the scenario fairly simple, we envisaged two potential sources of problems which we wanted to simulate. Either the customer was experiencing poor voice quality as a result of dropped packets within the wholesale network or as a result of excessive non real-time IP traffic in the access network.



*Note – the scenario is simplified by not including management of the the Local Access NW

Figure 43 VoIP Network Scenario

G.5.2 Shared remote access lab

Typically open source projects are successful when all of the contributors have access to the final product – through regular builds which can be downloaded and tested locally by all the participating parties. We conjectured that an open network management project seeking to bring several geographically dispersed groups together would benefit from a central laboratory test network which the co-operating teams could access to test and install their parts of the solution. In addition, we did not want to have to carry a substantial collection of hardware to Nice and wanted instead to be able to demonstrate the solution remotely through the internet. This requirement was met by building a test network at the University of Southampton which could be accessed remotely by all the parties through a VPN.

G.5.3 The test network

Figure 44 illustrates how the lab was realised in Southampton University. The following description refers to the node labels on each node in the diagram. All of the traffic nodes are at the top of the diagram while the nodes at the bottom of the diagram hosted the management solutions. In order to be sure that the simulation did not crash during a demonstration in Nice (an unlikely but potentially embarrassing prospect), we duplicated most of the systems in Southampton to ensure resilience.

A Cisco PIX VPN was used to provide remote VPN access into the lab environment. A Foundry switch was used to provide 5 VLANs each representing a separate sub network. This allowed the VoIP traffic to only follow a path which traversed the simulation network, while all of the management traffic and any remote configuration of the servers could be performed remotely using the VPN which could 'see' all of the underlying IP segments.

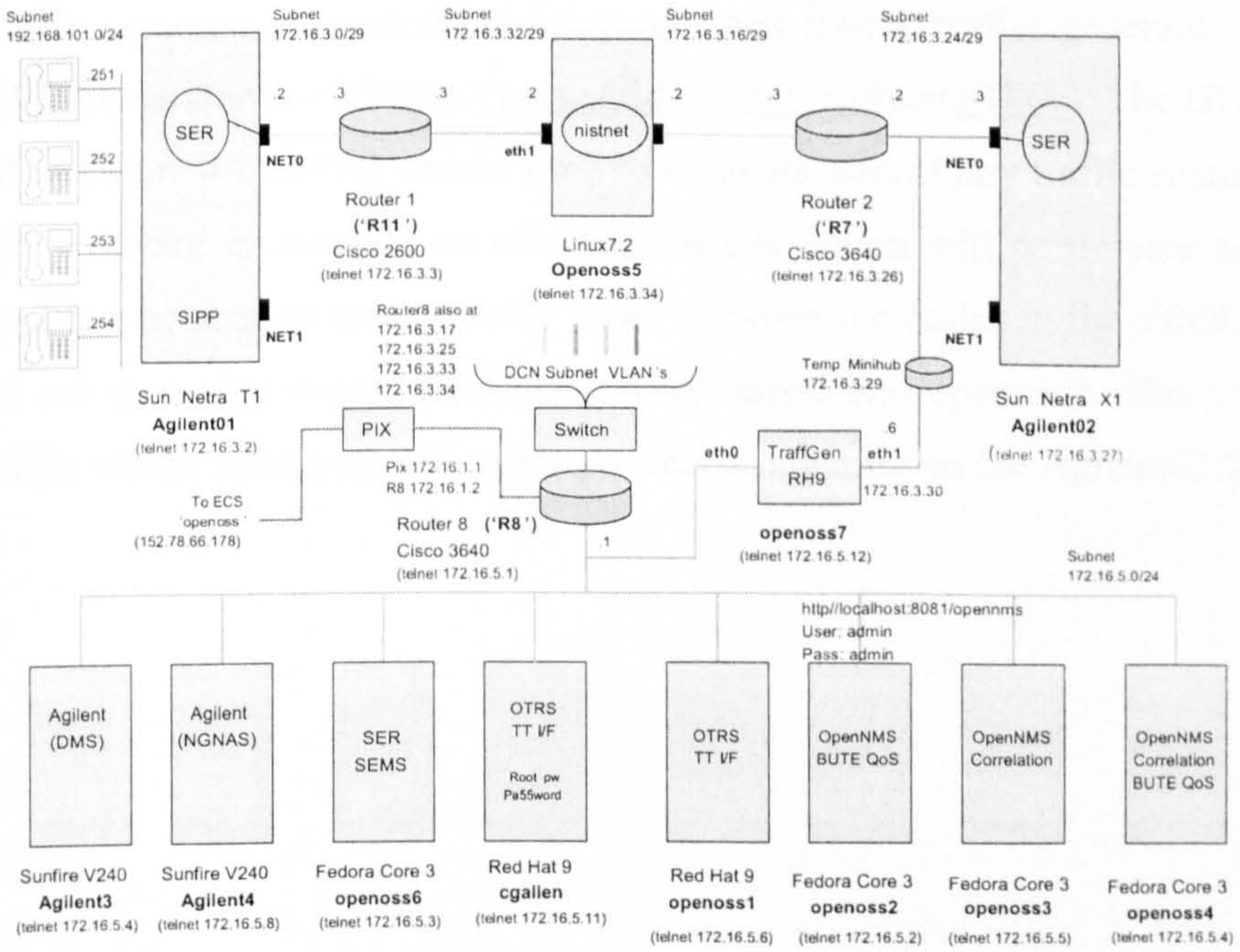
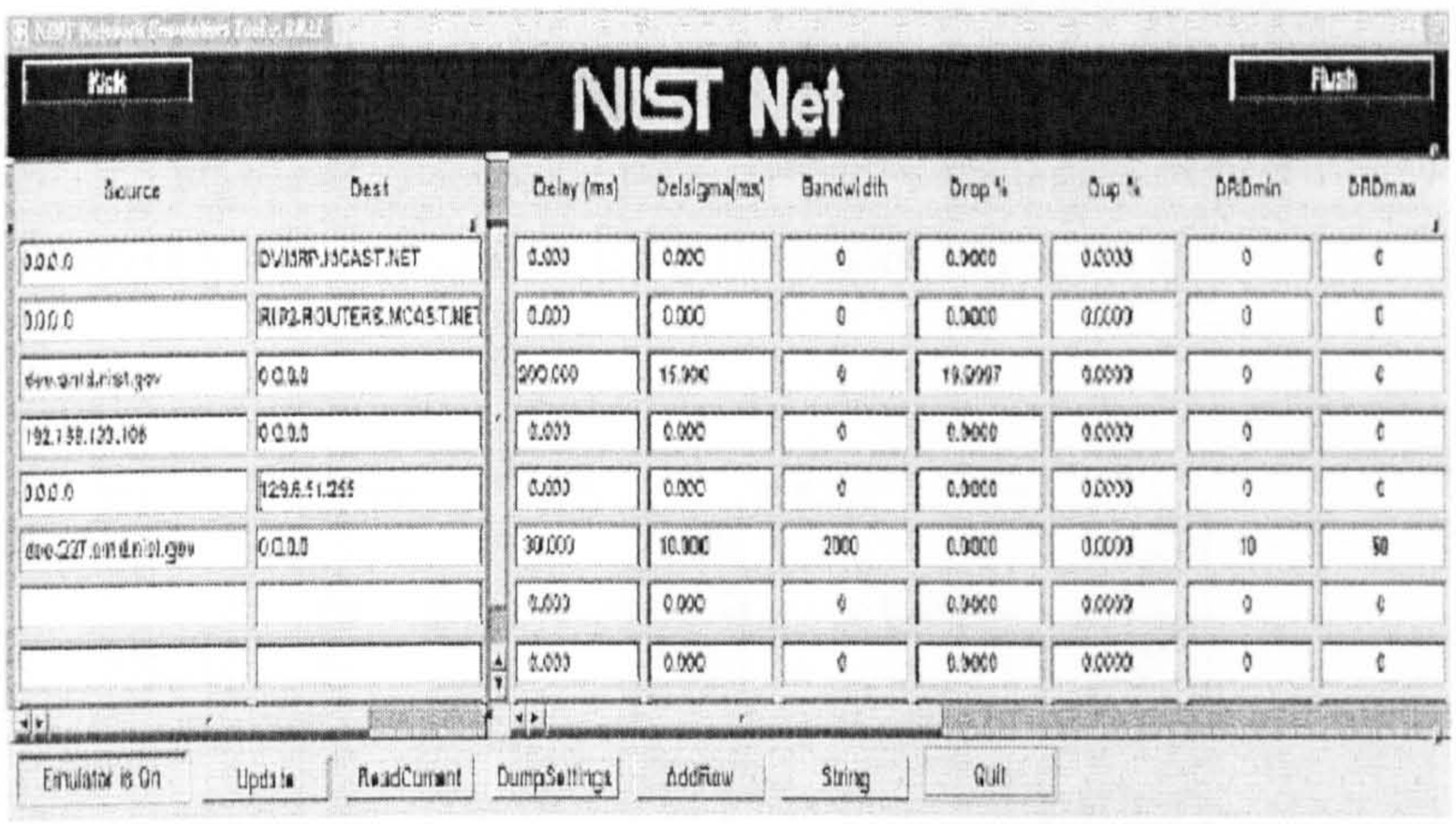


Figure 44 Simulation Network

Two routers 'Router 1' and 'Router 2', acted as the gateway routers in the scenario. All of the traffic traversed these routers so that traffic statistics could be monitored.

A network emulator on 'openoss5' could introduce controlled errors into the traffic between the routers, thus representing errors in the wholesale IP network. The network emulator used an open source program called Nistnet http://www-x.antd.nist.gov/nistnet/. The Nistnet GUI is illustrated in Figure 45. Nistnet runs on a Linux system configured as a router with two or more Ethernet ports and introduces known network errors according to a defined statistical pattern between two given network addresses.



NIST Net graphical user interface

Figure 45 Nistnet

The over-traffic situation was emulated using an open source traffic generator (Distributed Internet Traffic Generator D-ITG http://www.grid.unina.it/software/ITG/). The GUI for D-ITG is illustrated in Figure 46. D-ITG can be used to simulate almost any traffic scenario across a network. The software is installed on all of the nodes which will participate and a central control consol can be used to set up traffic flows between the nodes in the simulation. In our scenario, we set up a very simple flow between openoss5 and openoss7. This allowed us to introduce traffic which interfered with voice packets originating on the Agilent02 box.

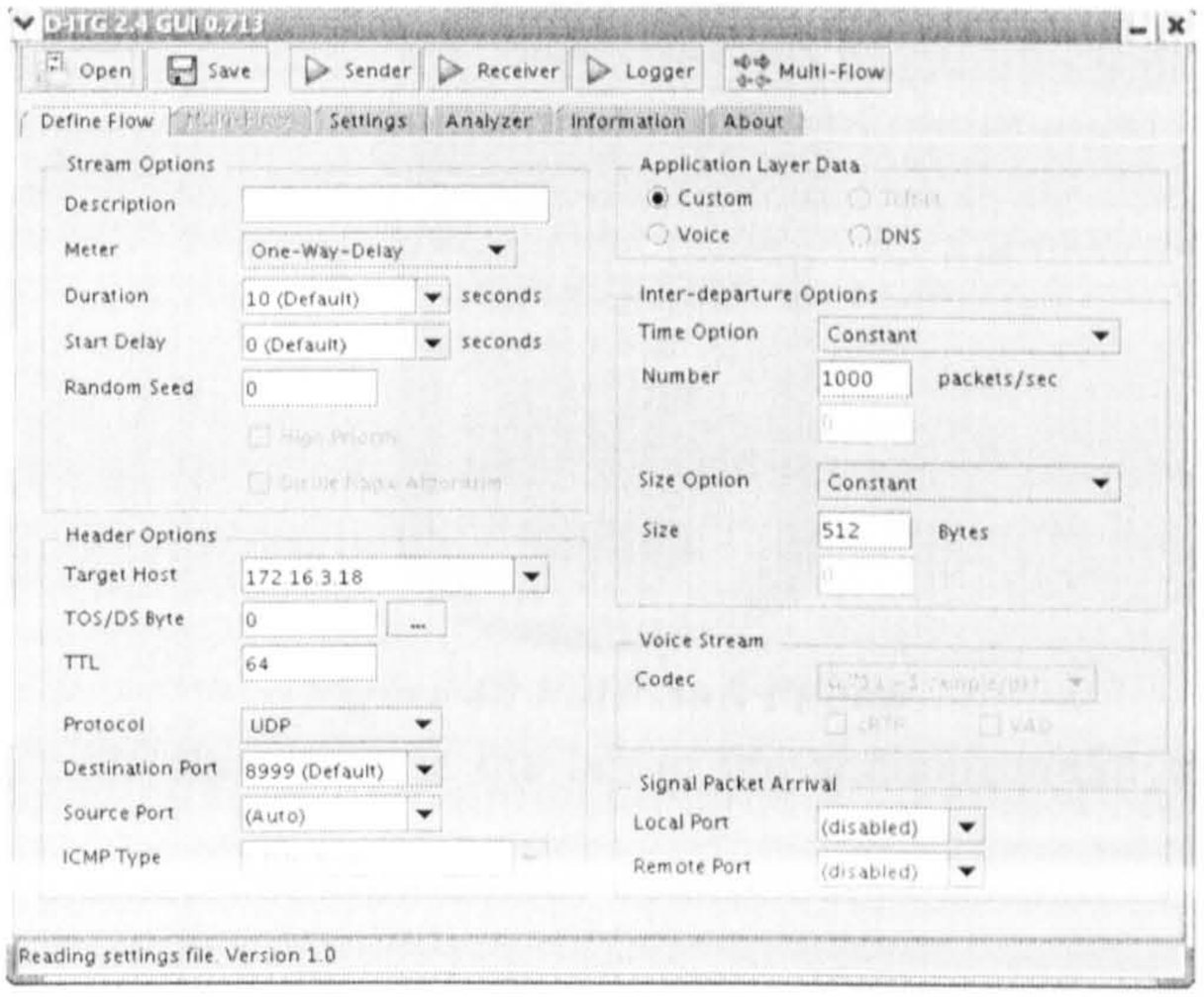


Figure 46 Distributed Internet Traffic Generator D-ITG

The routing of excess traffic through the lab is illustrated in Figure 49.

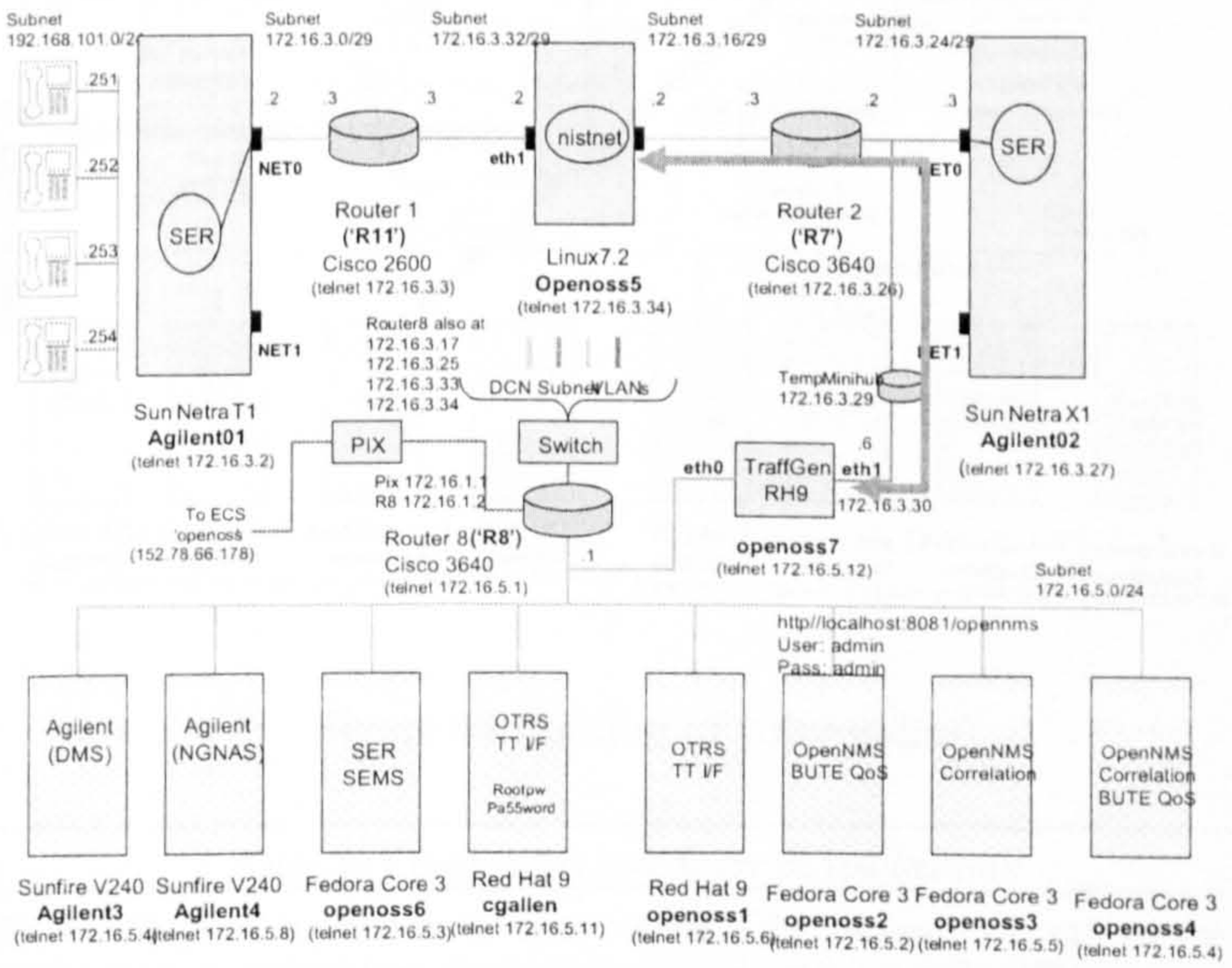


Figure 47 Routing of 'excess traffic'

G.5.4 VoIP

The VOIP solution used two open source VoIP packages and one free to download IP phone package

- SER Sip Express router http://www.iptel.org/ser/
- SEMS Sip Express Media Server http://sems.berlios.de/
- X-Lite - IP Phone http://www.xten.com/index.php?menu=products&smenu=xlite

The X-lite IP Phone which runs on any PC is illustrated in Figure 48. For the purpose of the demonstration, X-lite clients could connect to the lab through the VPN and initiate a call or dial a pre-recorded message running on the Sip Express media server. Both the calls and the recorded messages were be routed through the Nistnet Network Emulator

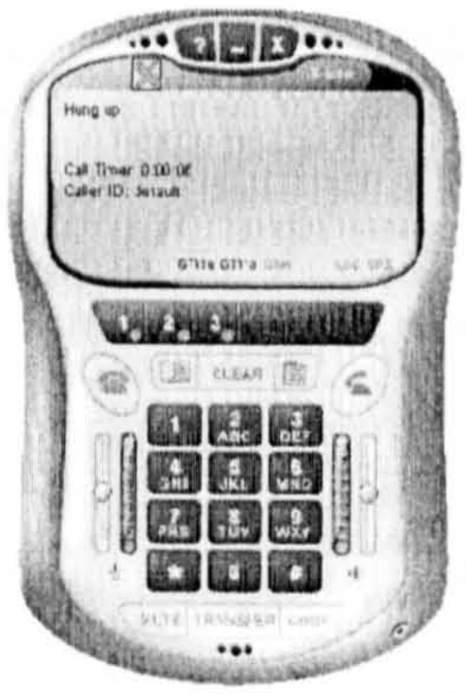


Figure 48 VoIP Soft Phone

The routing of voice calls from outside the lab to the announcement server is illustrated in Figure 49.

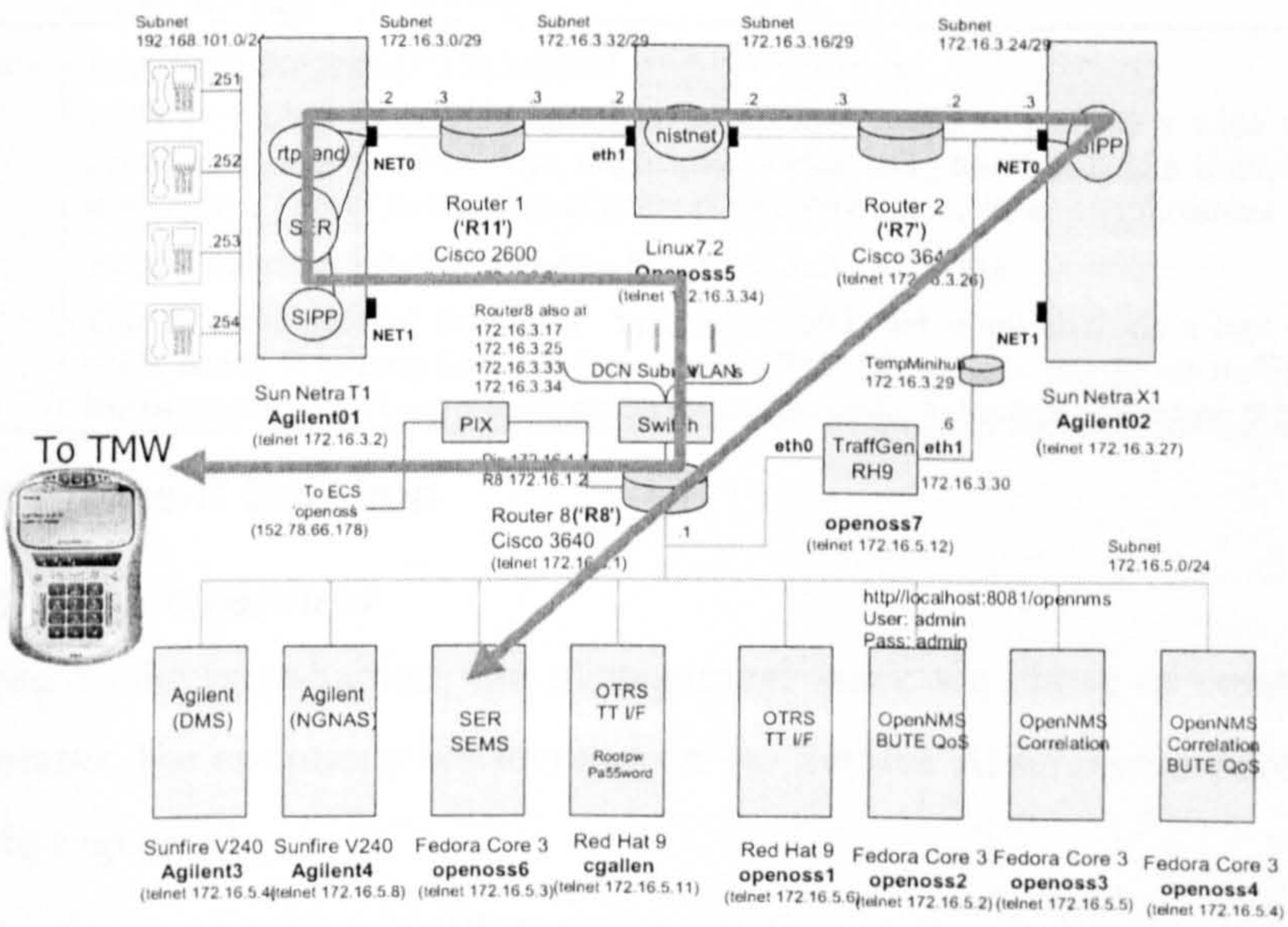


Figure 49 Routing of voice calls

Table 53 Components used to create test network	
SER	SER (SIP Express Router) is an open source (GPL license) SIP call server running on Linux http://www.iptel.org/ser/ (see also http://www.voip-info.org/wiki-SIP+Express+Router) SER is a high-performance, configurable, free SIP (RFC3261) server. It can act as SIP registrar, proxy or redirect server. Requirements: Linux/i386 (works with Red Hat) SER is in production with • SIPphone http://www.sipphone.com/ • TeleSIP http://www.telesip.net • Free World Dialup http://www.freeworlddialup.com/

Table 53 Components used to create test network	
SEMS	Sems is a free and extensible media server which helps you adding voice services to your VoIP system. Together with Ser (SIP Express Router), it turns every Linux box into a full-featured IP telephony system. Sems can be extended through plug-ins. They help you support more voice codecs or new applications. The plug-in Sdk is included for programmers who wish to create their own plug-ins. The following plug-ins are shipped with Sems: Voicemail: record messages and mail them. ISDN Gateway: support calls from and to the PSTN. Conferencing: connect people within a conference room. Announcement: plays an announcement. Echo: test module echoing your voice. The following audio codecs are shipped with Sems: G711.u G711.a GSM 06.10 iLBC (development version) speex (development version)
NISTNet	http://snad.ncsl.nist.gov/itg/nistnet/ NIST Net allows a single Linux box set up as a router to emulate a wide variety of network conditions, such as packet loss, duplication, delay and jitter, bandwidth limitations, and network congestion. This allows testing of network-adaptive protocols and applications Requirements :A Linux installation (2.0.xx, 2.2.xx or 2.4.xx). One or more network interfaces. Typically, NIST Net is installed on a box with two Ethernet cards which is routing between two subnets. This allows it to munge all traffic flowing between the two networks. It can also be set up on an end node, to munge traffic into that node.

G.6 Management Solution

G.6.1 Solution overview

As described in the introduction, the management stack we chose to create was for a VoIP service operator. For simplicity we focused on the Service Assurance aspects of the eTOM as illustrated in Figure 50.

The characteristics of such a VoIP operator are similar to a large enterprise who buys their connectivity from an external supplier. Thus from a management perspective, the VoIP network appears as a distributed set of sites, each having a locally connected set of nodes which are geographically connected through an unmanaged Wholesale IP offering (IP VPN). The operator can monitor the ingress and egress traffic but cannot directly see what is happening within the wholesale provider's network.

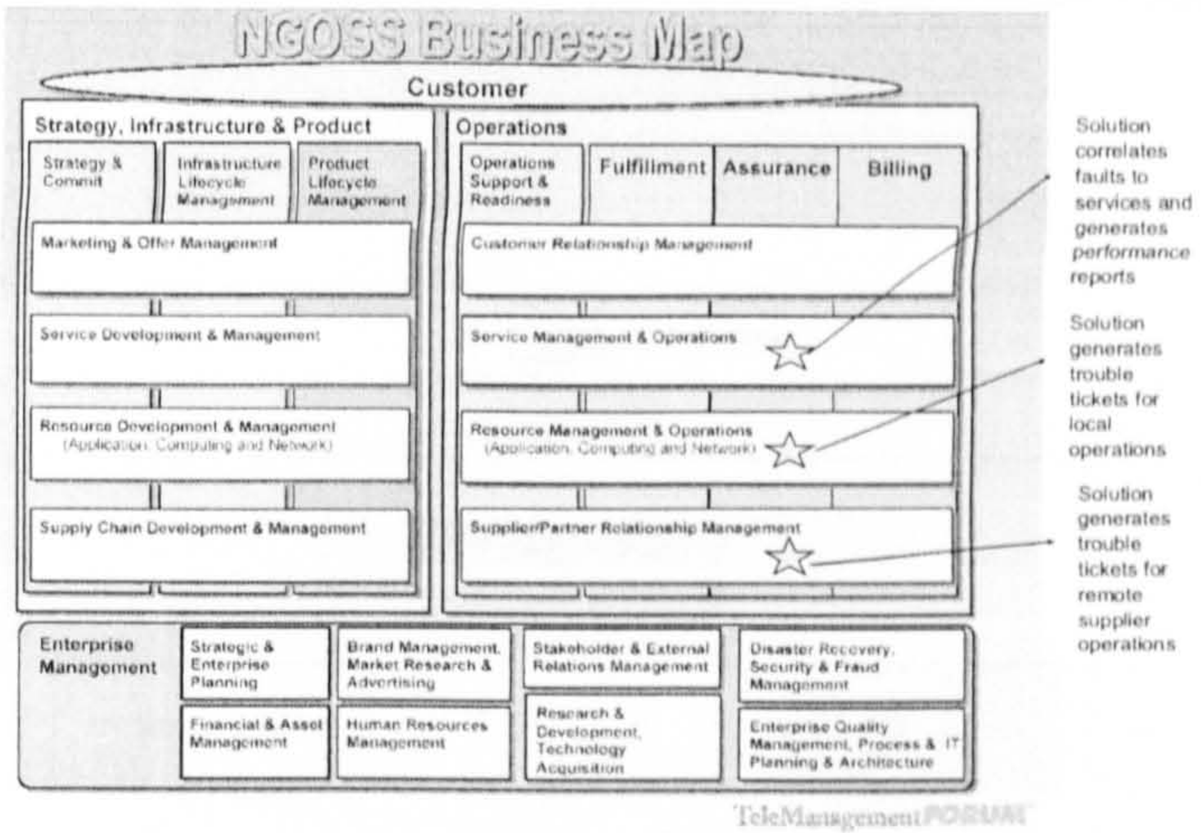


Figure 50 Open OSS Catalyst mapping to eTOM

In this scenario, the VoIP operator must manage their customer's experience by monitoring performance of their own equipment and also by monitoring the quality of the interconnecting VPN against the SLA they have agreed with their wholesale service provider. The conceptual OSS stack chosen to do this is illustrated in Figure 51.

The raw traffic at the ingress and egress points of the network can be measured by the gateway routers. More sophisticated RMON measurements and threshold alarms can be set up on these gateways if desired. The VoIP service components (running on Linux platforms) can also be monitored. All of these measurements and threshold events are collected using SNMP by the Data Traffic Monitor.

Monitoring raw traffic may tell us there is a problem but of itself tells us very little about the functioning of the VoIP service. To understand how successful VoIP calls are, it is necessary to monitor the VoIP signalling using a SIP signalling monitor. The signalling monitor gathers statistics on the progress of calls through the network. This can tell us if there is a high incidence of dropped calls which could be a result, for instance, of the SIP servers being mis-configured (e.g. numbering plan is set up incorrectly). Most usefully, it also obtains statistics from the end clients as to the number of out of sequence (i.e. dropped) packets during a call which has a direct bearing on the voice quality. By monitoring at key points in the network it is possible to begin to tell where VoIP problems are occurring and target the solution.

In many cases multiple thresholds or repeating alarms are generated related to a single fault and therefore a correlation solution which can filter these alarms down to single trouble tickets is a useful capability. Depending upon the service provider's business process, several tickets could be created when dealing with a single fault – Customer problem reports, Network problem reports and External problem reports to send to the Wholesale IP provider. Thus the correlation solution should be flexible enough to allow rules to be created for the generation of trouble tickets in order to suit the needs of the service provider's business process.

This solution is a relatively standard OSS stack and could be readily realised using commercial products. However with the open OSS catalyst, we wanted to investigate how much of this

solution could be realised using a mixture of COTS and open source components integrated using OSS/J.

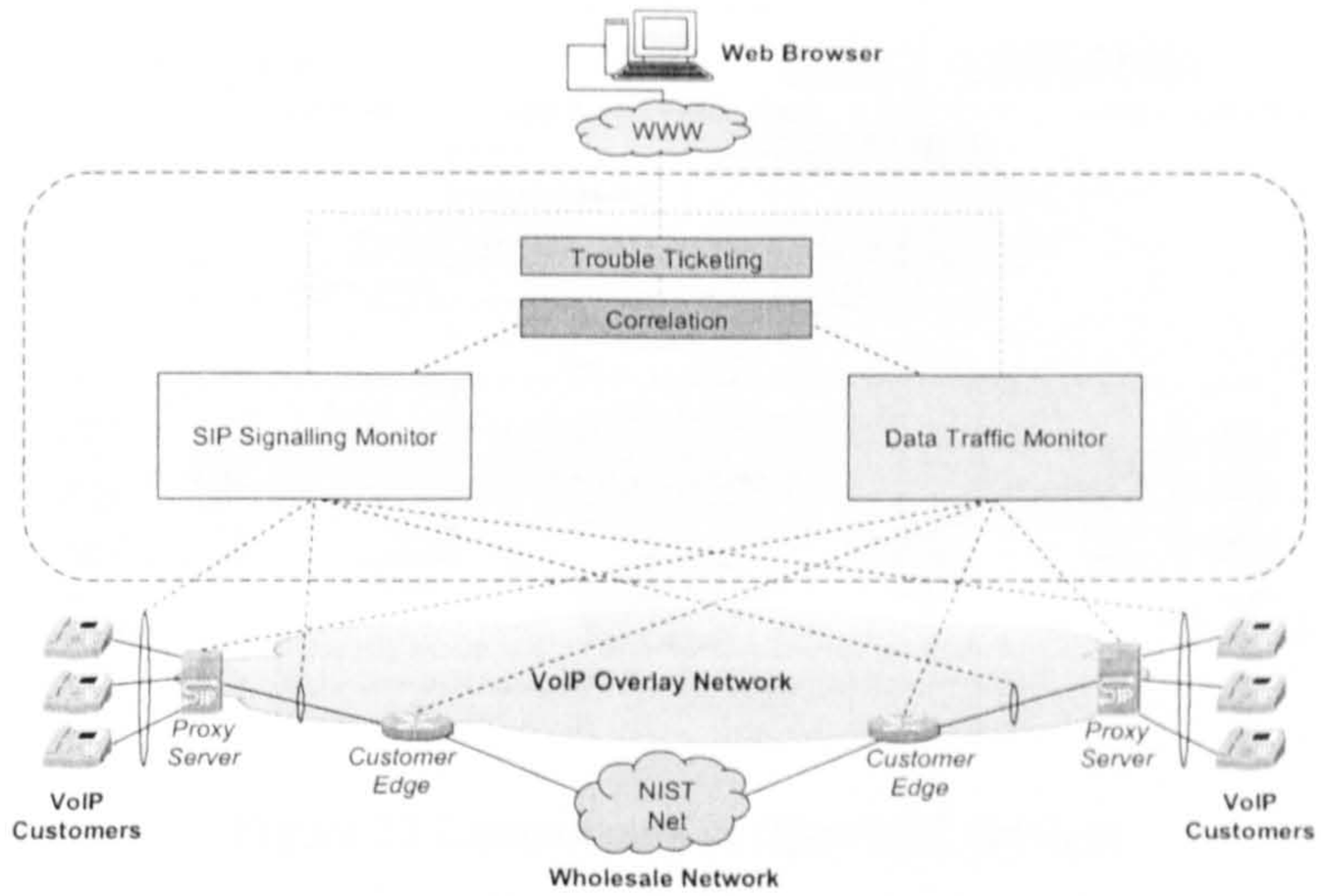


Figure 51 Conceptual OSS stack

G.6.2 OpenOSS Catalyst realisation of the OSS stack

Figure 52 illustrates how this solution was realised in the OpenOSS catalyst.

The SIP Signalling monitor was realised using a COTS product from Agilent called NgnAS (*Agilent 2002*). Agilent added an experimental web interface and OSS/J adaptor to the standard product in order to integrate into the solution.

The Data Traffic Monitor was realised by the Budapest University of Technology and Economics (BUTE) and the University of Southampton (UoS) with British Telecom based upon an open source enterprise network management product called OpenNMS.

BUTE created an OSS/J Performance Management adaptor for OpenNMS and a Performance Management portlet running on an open source portal server (JBoss Portal Server) to access the PM data.

UoS created an OSS/J Fault Management adaptor for OpenNMS which fed alarms into a simple correlation engine based on the Drools open source inference engine. The inference engine was able to create trouble tickets which were fed through an OSS/J TT interface into an open source trouble ticket application called OTRS.

It is important to note that the objective of the exercise was to prove that an open source OSS stack was realisable and that this could be usefully integrated with COTS solutions using OSS/J interfaces. We did not have time to create a fully realised solution with complete OSS/J implementations and sophisticated correlation rules. However we were able to show that this could usefully be achieved given more time and resources.

In the following sections we will describe the contribution of each group in more detail.

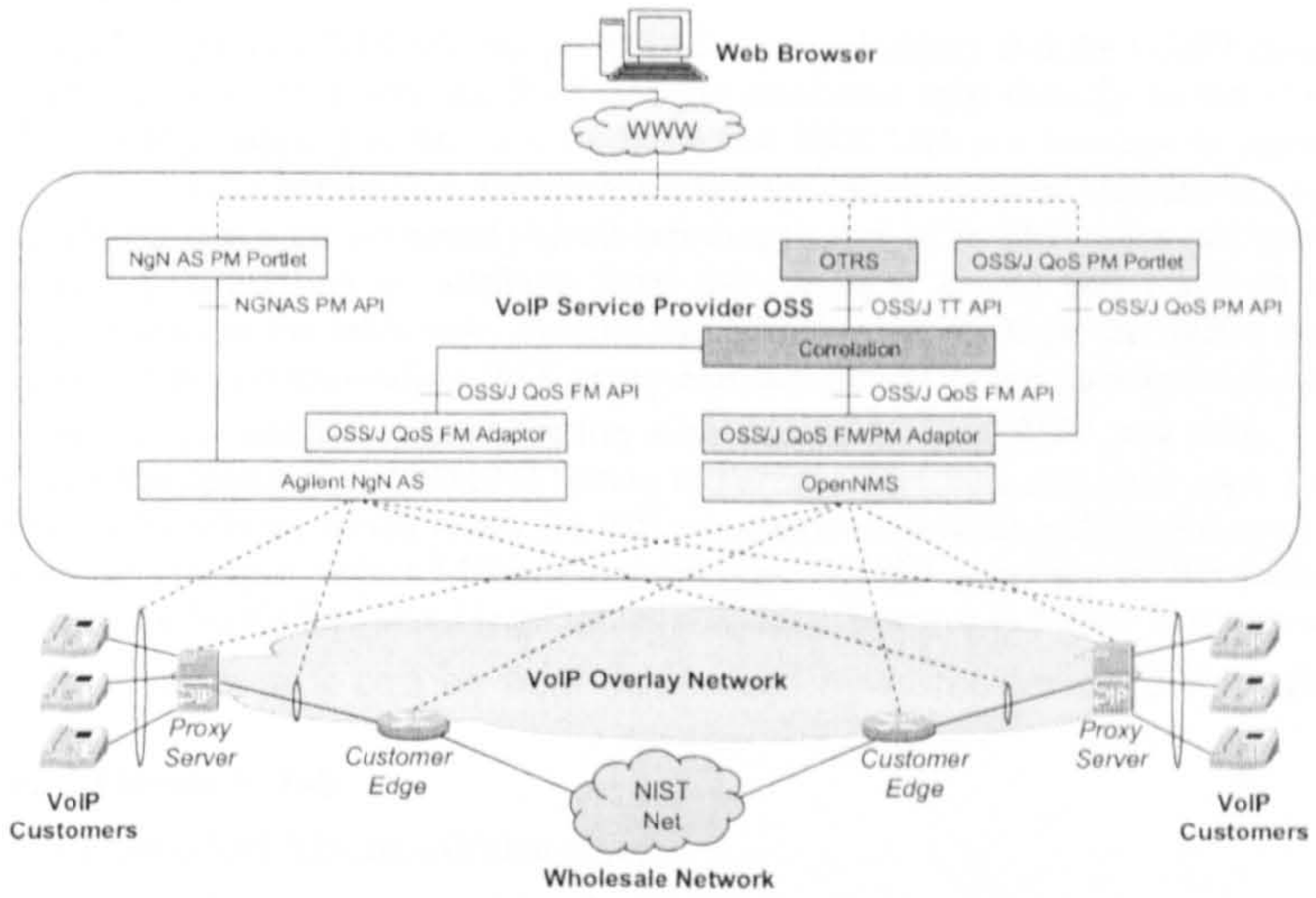


Figure 52 Components in OpenOSS catalyst

G.6.3 VoIP Management (Agilent)

In this section we describe the work done by Agilent Technologies to create the SIP Signalling monitor using COTS product from Agilent called NgNAS. Agilent added an experimental web interface and OSS/J adaptor to the standard product in order to integrate into the solution.

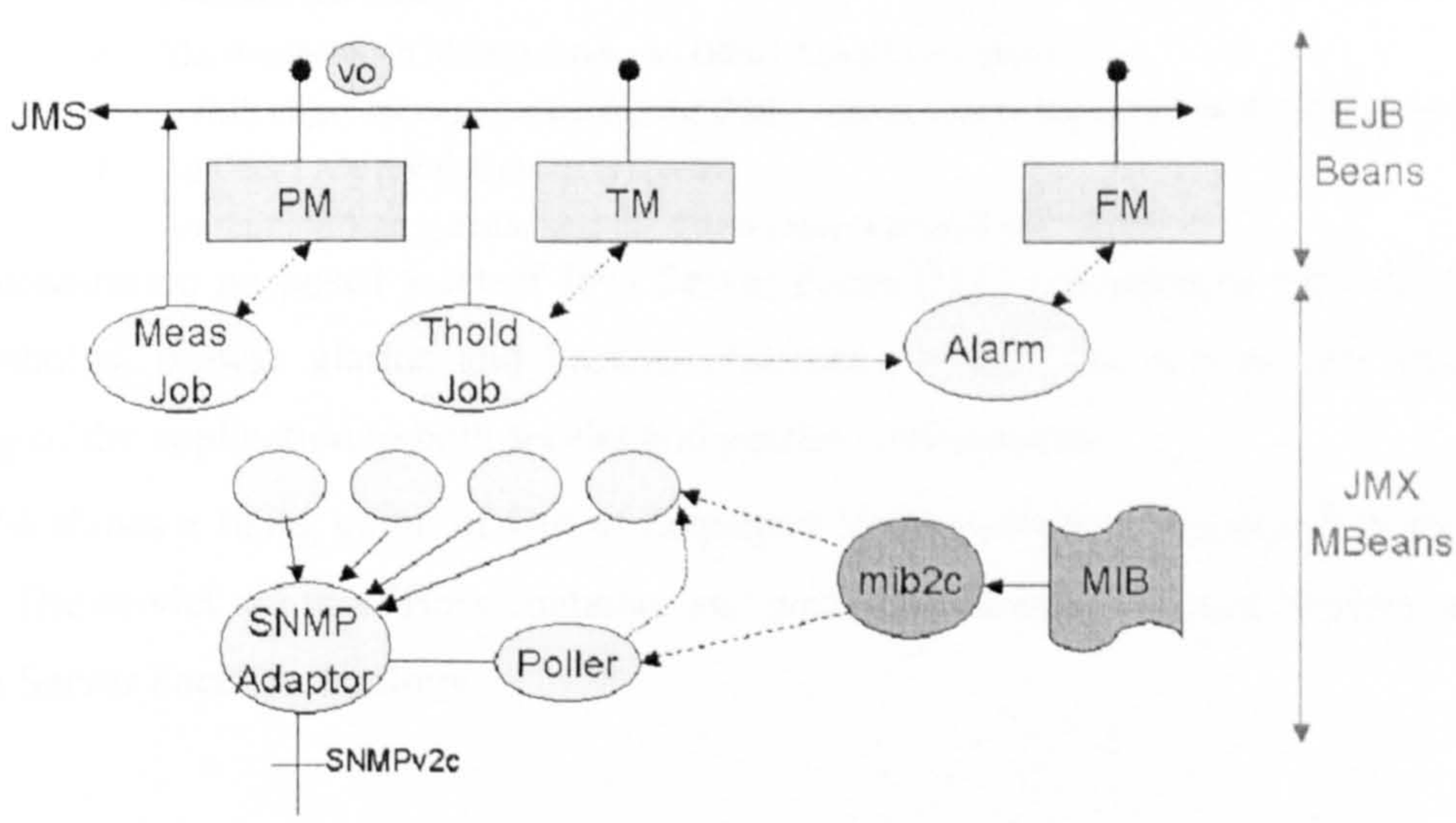


Figure 53 Agilent architecture for OpenOSS

Agilent Technologies developed an OSS/J QoS interface for the NGN Analysis System as its contribution to the OpenOSS catalyst project. This is illustrated in Figure 53

The interface logic is implemented is as three Enterprise Java Beans (EJBs) for the fault (FM), threshold (TM) and performance (PM) interfaces respectively. All other components in the system are implemented as JMX MBeans (defined in JSR-002). The MBean component model allows the dynamic access and creation of object attributes, access to object operations and an asynchronous notification mechanism based on the observer / notify pattern. There are several types of MBeans in the system

1. OSS/J Entity Representations: The PM, TM and FM interfaces manipulate the conceptual entities measurement job, threshold job and alarms respectively. These MBeans represent those OSS/J entities and their lifecycle is managed by the EJBs
2. Observed objects: There is a JMX MBean per OSS/J managed object and the OSS/J object name maps directly to the JMX object name. Similarly the JMX MBean attributes map directly to the OSS/J observed object's attributes. This strong mapping means it is trivial to use JMX MBeans services to implement complex OSS/J capabilities e.g. reflection, persistence, serialization and the use of monitor MBeans to implement thresholds. In the current implementation the observed objects represent a row in an SMI table and thus implement an SNMP proxy function e.g. requesting an attribute from the managed object will result in an SNMP get to the corresponding column in the table row. Passing the SNMP MIB through the mib2c compiler with a custom backend to generate the corresponding JMX proxy MBeans generates the managed objects java classes.
3. Agent Services: At present all communication with agents is via SNMP and there is a SNMP adaptation MBean based on the open source SNMP4J library to perform gets, sets and send traps. Each SNMP table has a corresponding poller MBean which walks the table to discover inserted and deleted rows and create and delete the corresponding managed object MBeans accordingly. For the purposes of the catalyst a SNMP wrapper agent was written for NgNAS to avoid legal issues with interfacing to 3rd party libraries used in the product.

The operation of the framework can be best described by considering 2 key use cases

1. **Creation of a Threshold Job**
    a. Parse the OSS/J ThresholdValue object
    b. Create an MBean to represent the threshold job and register it
    c. Locate the observed object MBean and attribute to be monitored
    d. Create a JMX Monitor MBean that observes the observes object and load the threshold information from the ThresholdValue object
    e. Bind the threshold job MBean as the listener for threshold exceeded notifications from the monitor MBean
    f. Start the monitor MBean which in turn will create a thread to poll the observed object
2. **Threshold Exceed and Alarm Raised**
    a. On exceeding the programmed threshold the monitor MBean will send a JMX notification to the threshold job MBean
    b. The threshold job MBean creates an OSS/J AlarmEvent object
    c. A JMS object message containing the OSS/J AlarmEvent is dispatched on the JMS alarm topic
    d. An OSS/J AlarmValue object is created
    e. An alarm MBean is containing the AlarmValue is created and registered

For demonstration purposed a set of Java Server Faces (JSF) components were developed to set thresholds, browse alarms and browse observed objects. The use of JSF allowed the targeting of the application to both servlet and portlet environments.

Figure 54 shows a rising count of Out of Sequence VoIP packets as displayed on the Agilent Servlet. The servlet ran in a JBoss container and was realised using Apache implementation of the Java Server Faces technology.

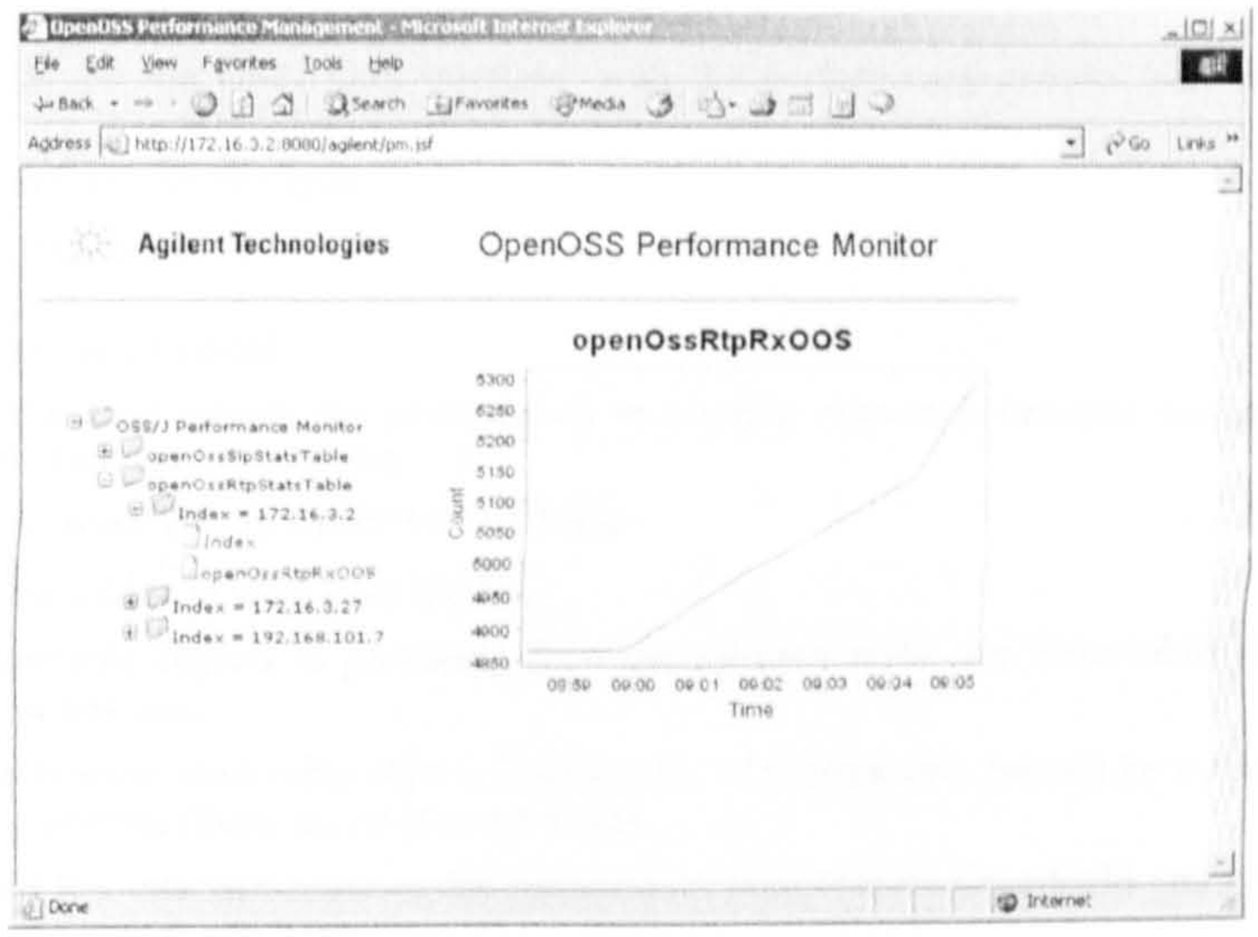


Figure 54 Agilent GUI for VoIP data

Table 54 COTS components used in the proof of concept		
ID No	Name	Details
(1)	NgNAS	Agilent Next Generation Network Analysis System See http://we.home.agilent.com/USeng/nav/-536885380.536882660/pd.html Key Features (from data sheet) • carrier-class monitoring solution with VoP/Internet Offload and SS7 real time signalling correlation• Protocols supported include MGCP, SS7, SGCP, Packet Cable NCS, SIP,H.323, IPDC, RADIUS and Nortel ASPEN; • Detailed call flow records (CFRs) for each call contain all signalling messages, addressing and routing information, quality metrics (delay, jitter), response codes and more; • Service measurement records (SMRs) are stored in the database for each and every call. SMRs contain detailed service and performance analysis results, providing information on Answer Delay, Call Duration and Dial Tone Delay, dropped packets; temporary and permanent failure cause codes, Dial Delay and Post Dial Delay, answer seizure ratio and more; • All signalling feeds are non-intrusively monitored in a failsafe manner ensuring network connectivity; NgNAS has been in production for over 3 years and is now seeing a ramping up of demand based on the growth of VoIP Currently NgNAS product has a Java GUI which supports multiple power users but want to move to support 100's of user in an Enterprise infrastructure. NgNAS team is currently in process of re-factoring internal services as EJB's which will be accessed by JSF based web display technology. This move to J2EE can be seen as a first step towards using OSS/J to provide standard interfaces Agilant want to understand better the opportunity to use Portlets and OSS/j to integrate system into a wider enterprise environment. They are also interested in understanding the integration of the system with external Workflow.

G.6.4 Performance Management (BUTE)

In this section we describe the work done by the Budapest University of Technology and Economics to create an OSS/J Performance Management adaptor for OpenNMS and a Performance Management portlet running on an open source portal server (JBoss Portal Server) to access the PM data.

G.6.4.1Use Case for the QoS PM

The following use case has been developed and described in the interaction diagram Figure 55.

There are four actors in the use case:

1. User: the end user who monitors and view performance reports via web interface provided by the JBoss Portlet.
2. JBoss Portlet: the portlet located in JBoss Portlet server. The portlet retrieves performance information and provides them to users.
3. OSSJ QoS PM i/f: the OSSJ QoS interface, with the performance monitor part developed to work with OpenNMS.
4. OpenNMS: network monitor agent.

Description of the use case:

a. The User logs in to the portal.
b. After login, the User selects the performance monitoring operation between available operations (trouble tickets, SIP, performance monitoring ...).
c. Network nodes monitored by OpenNMS are listed.
d. The User selects a node to view from the list.
e. A list of observable objects is presented from the selected node. An observable object is a performance measurement in this case.
f. The User selects some observable objects from the list of performance reports he wants to view. Furthermore, the monitoring interval (from, to) must be provided.
g. The OSSJ QoS PM interface retrieves the performance reports from OpenNMS and gives back to the portlet. The User can see it on the web interface.

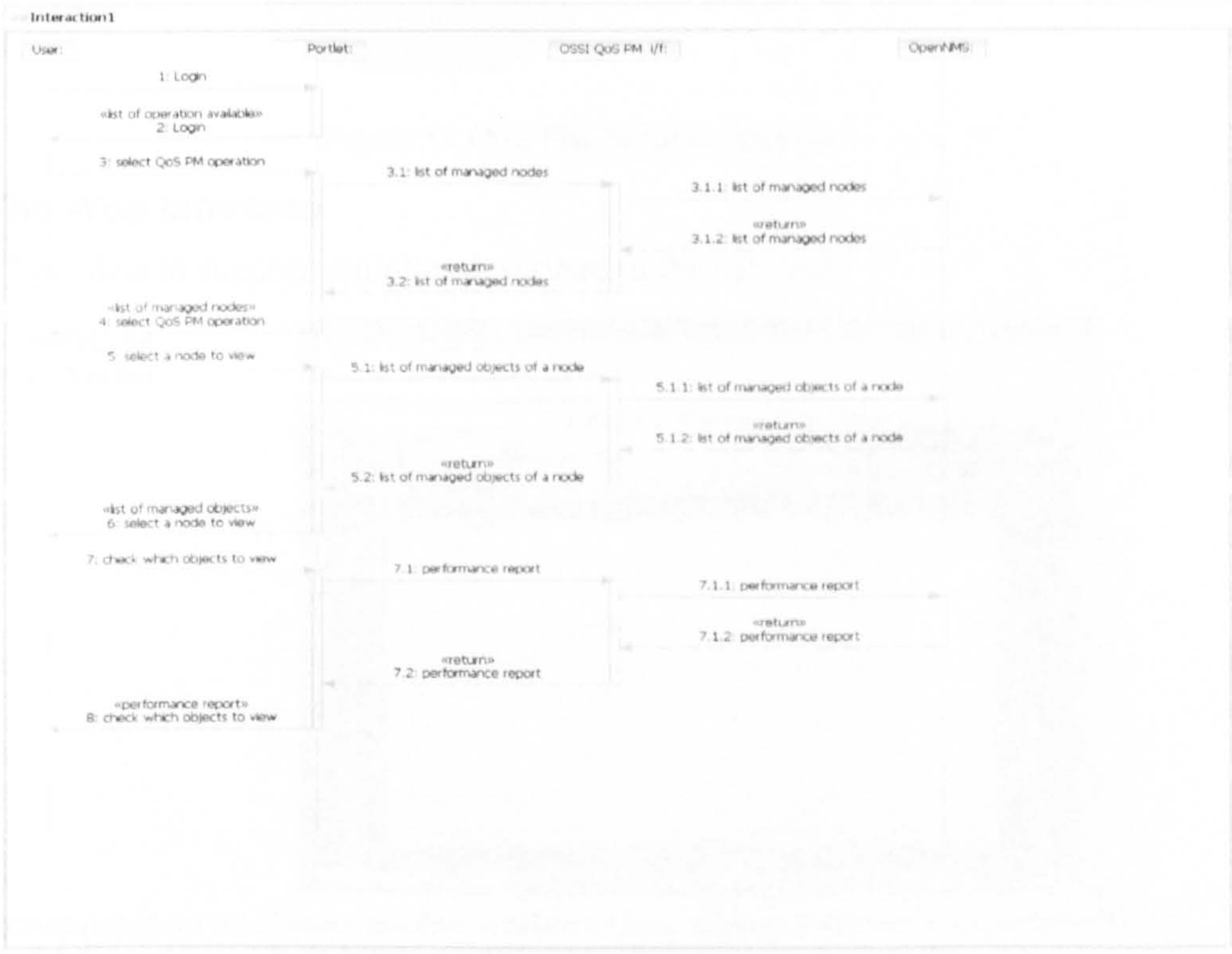


Figure 55 QoS portlet interaction diagram.

G.6.4.2The Technical Solution

Figure 56 illustrates the Qos PM Solution design.

OpenNMS stores node configuration data and event log records in PostgreSQL and uses flat files to store performance data for performance reports. The performance data is managed in OpenNMS using the JRobin component (www.jrobin.org).

The Qos PM solution accesses the databases directly to provide performance data across the OSS/J interface. JRobin is used to extract data from flat files and JDBC to access the PostgreSQL database and make it available through the OSSJ QoS PM interface.

A performance management portlet was written to access the OpenNMS data through the OSS/J interface. The portlet can in theory run on any JSR168 portal server.

However in our demo, we used the open source JBoss portal server (http://www.jboss.com/products/jbossportal) and it has also been tested on the open source Gridsphere portal server (www.gridsphere.org)

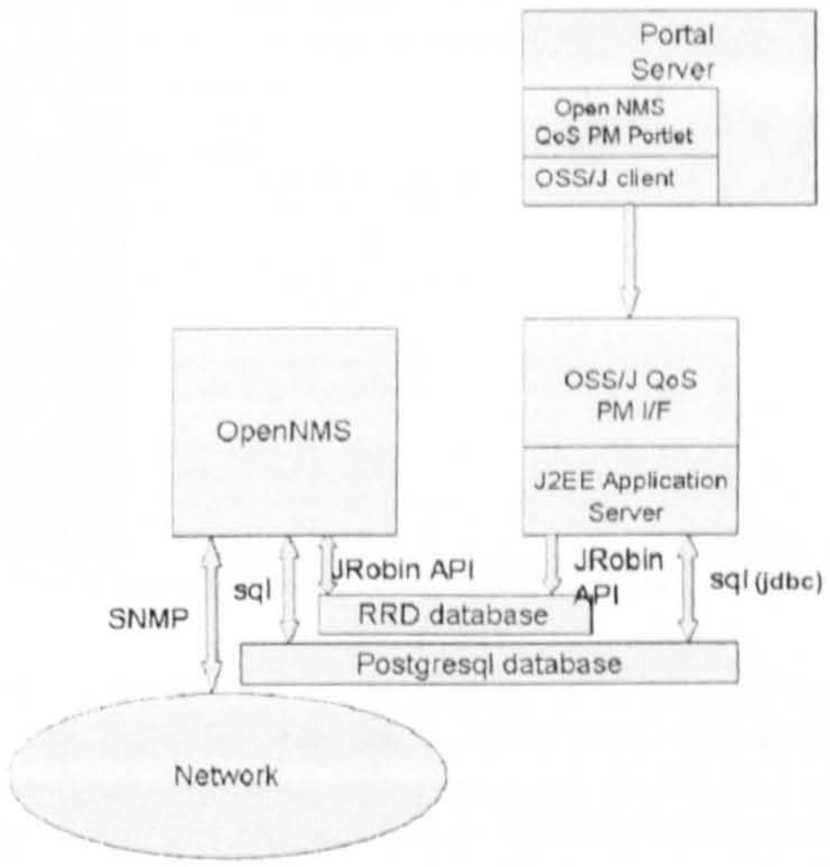
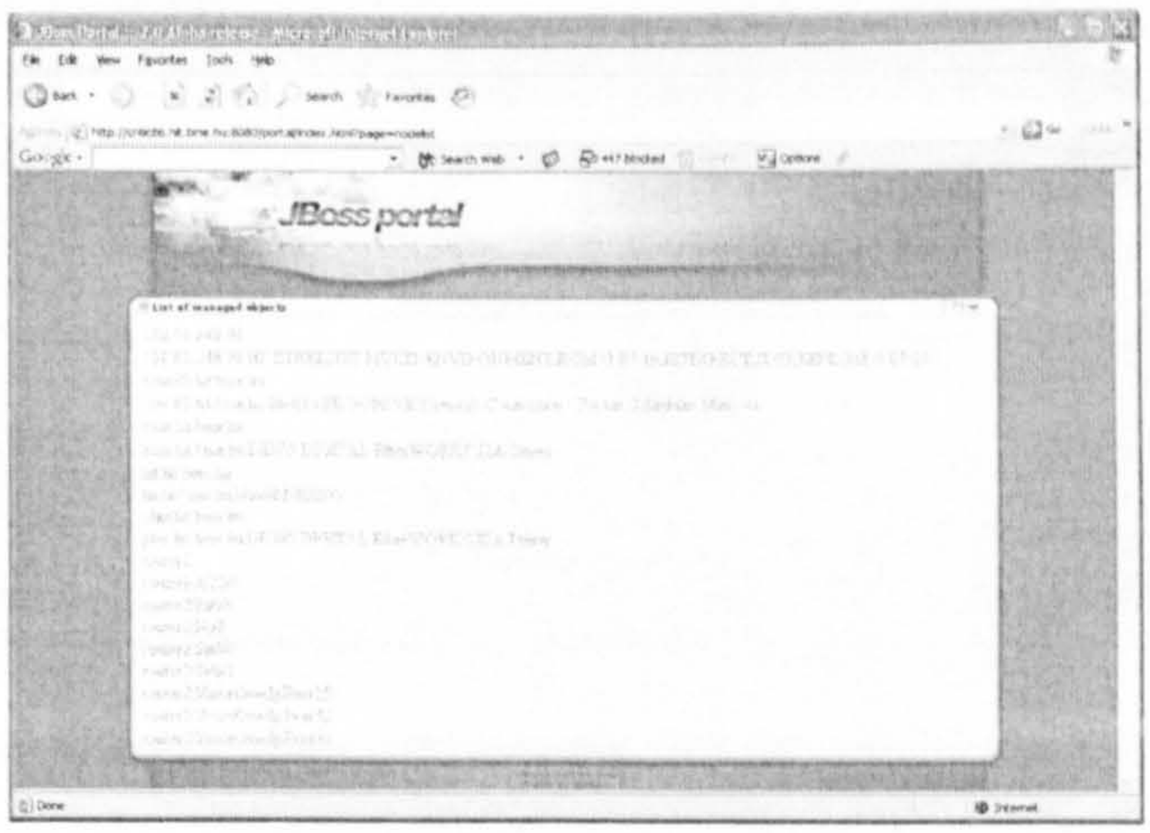


Figure 56 QoS PM Solution Design.

G.6.5 The Web Interface

We have 3 portlets to support activities described in the use case.

a) **List of Nodes and Interfaces Portlet**

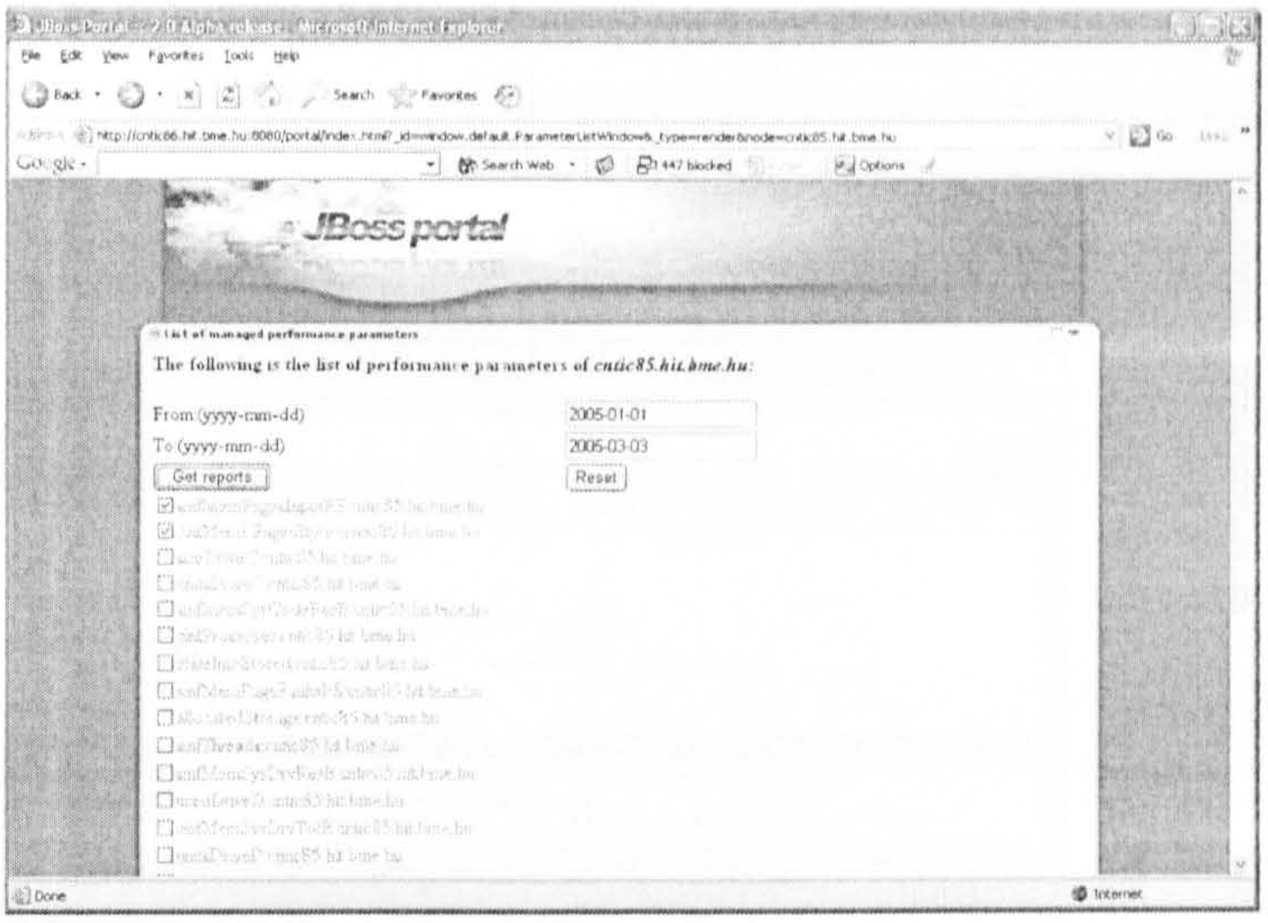


This portlet provides a list of nodes and their interfaces. Only nodes are listed which support SNMP. The interfaces belonging to a node are listed immediately after the node.

For example, in the captured window above, node 152.66.248.98 has one interface, namely HP ETHERNET MULTI-ENVIRONMENT.ROM while node router2 has several interfaces: AT2/0, Fa0/0, ...etc.

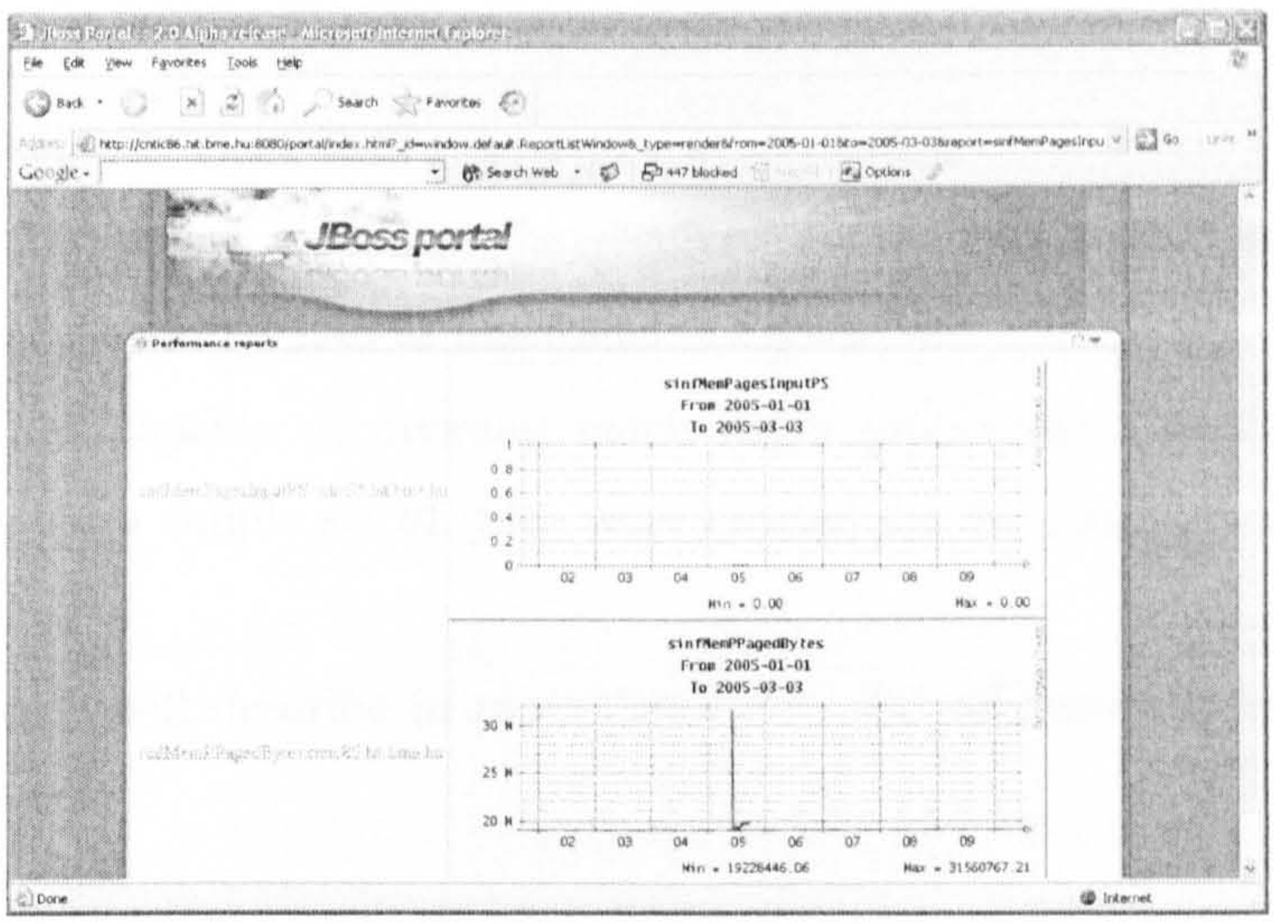
If we want to view a list of performance parameters of a node or an interface, just click on that node or interface. Then the portlet shown in the following will be displayed.

b) **List of Parameters Portlet**



This portlet provides a list of performance parameters of a chosen node or interface. Beside the performance parameter on the right we can see a check-box indicating whether it is selected for performance reporting or not (default is not). The reporting interval can be entered in the "From" and "To" fields. Finally, clicking on the "Get reports" button will display the next portlet shown below.

c) **List of Reports Portlet**



This portlet provides the performance report graphs of the parameters selected in the previous window.

G.6.6 Fault Management & Correlation (Southampton)

In this section we describe the work done by the Southampton University to create an OSS/J Fault Management adaptor for OpenNMS, correlation and Trouble ticket interface. This section provides an outline of the work done. Full descriptions of the work are found in the student project reports [3]. [4].

G.6.6.1Basic function of the demonstration

In the catalyst we demonstrated the correlation engine working by using D-ITG to generate excess traffic across router 2 as illustrated in Figure 47. The router was configured so that it would generate a series of RMON traffic threshold traps for the input and output measurements on the ports where the traffic was presented. These traps were monitored by

OpenNMS as illustrated in Figure 57 and passed as alarms to the correlation engine through the OSS/J interface.

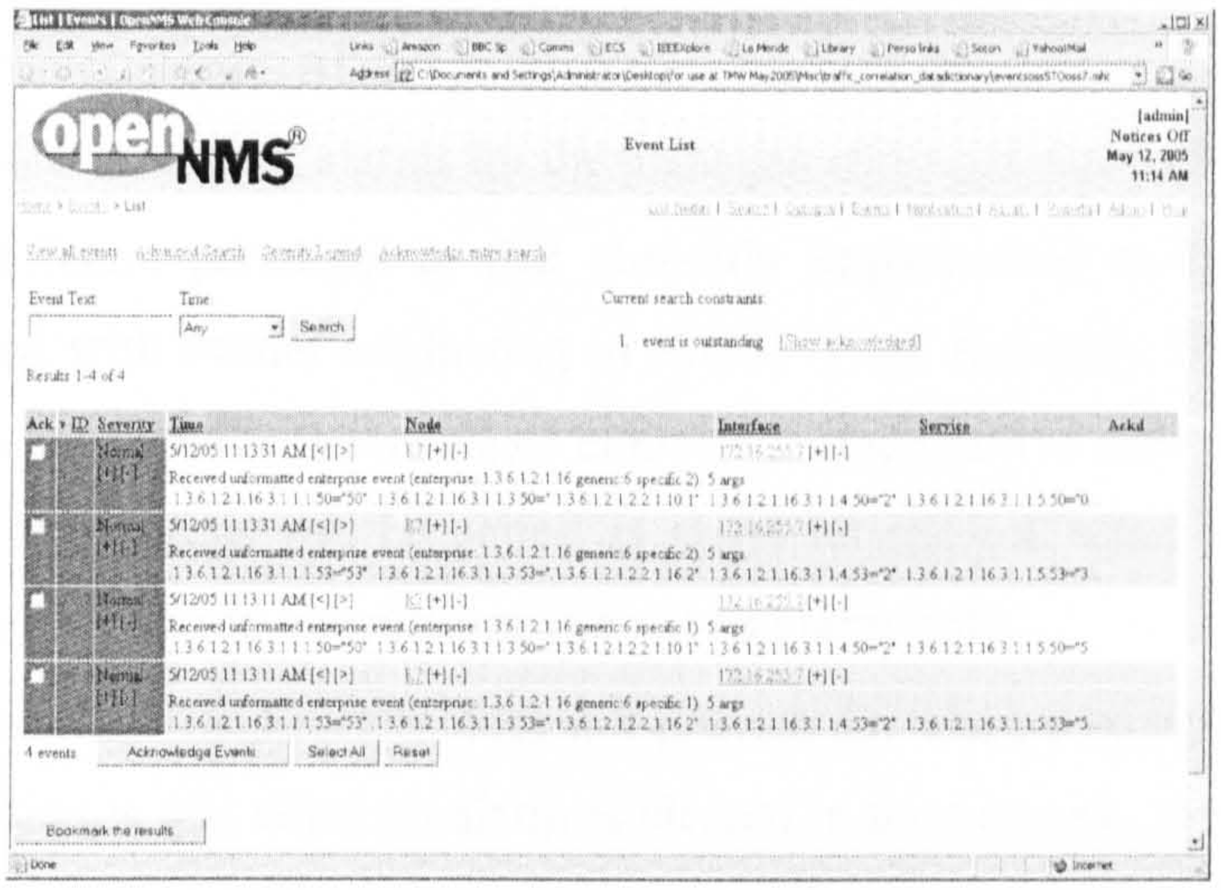


Figure 57 OpenNMS alarms prior to correlation

The rules in the correlation engine examined the received alarms and created appropriate trouble tickets which were passed to OTRS as illustrated in Figure 58.

The correlation engine was capable of creating much more sophisticated trouble tickets depending upon the rules but a simple set of rules were created for the Catalyst simply as a proof of concept.

In the following sections we will describe in more detail how the solution was constructed using open source components.

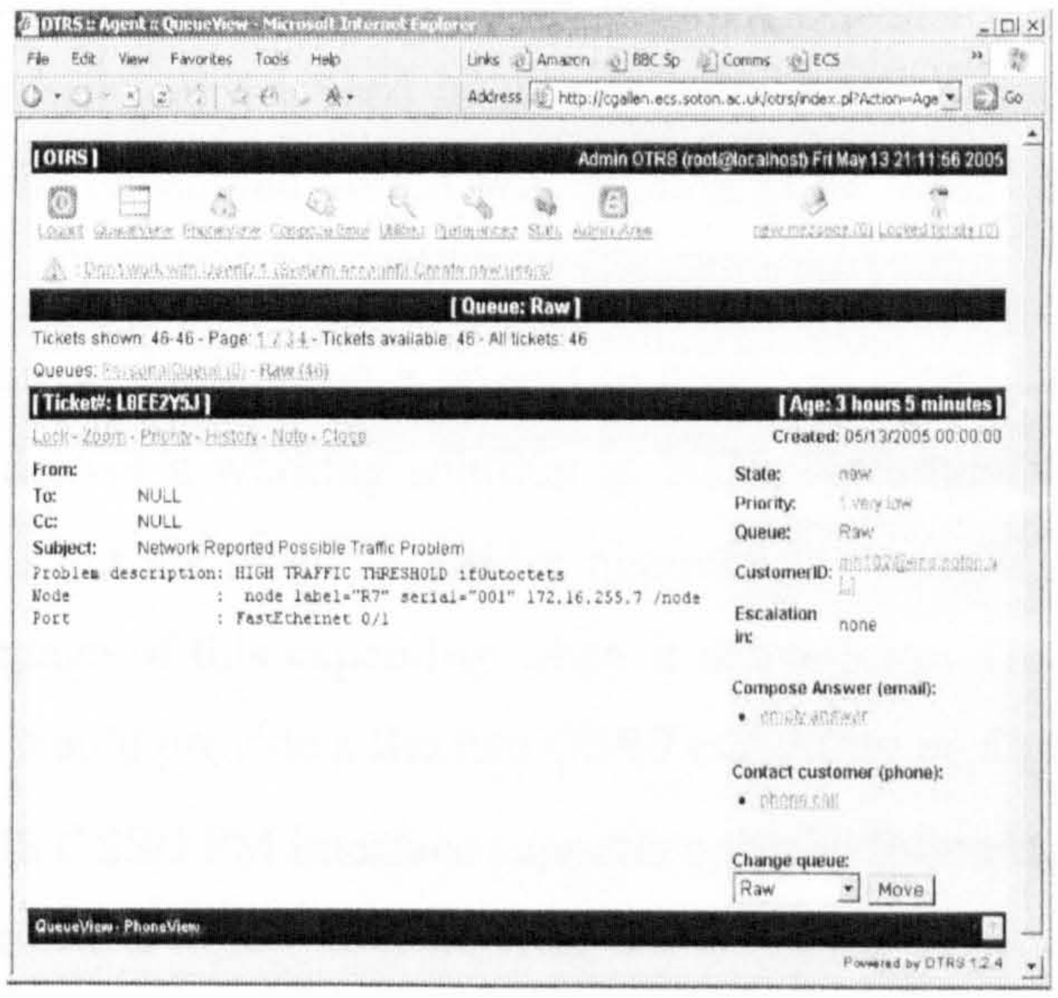


Figure 58 Correlated trouble ticket

G.6.6.2OpenNMS OSS/J QoS FM Interface

The OpenNMS OSS/J QoS FM Interface is explained more fully in [3] but in outline the system works as follows.

As a telecoms standard, the OSS/J QoS FM interface is primarily concerned with Alarm State rather than with a list of events. At any point of time the OSS/J interface is expected to provide a snapshot of the current state of alarms for the managed entities it represents.

This is quite a different paradigm to that currently implemented in OpenNMS which is primarily concerned with events originating in SNMP Trap messages. SNMP traps indicate that something has happened in the managed entity. It is important to note that not all traps are related to alarms. Some traps will be paired as alarm raising and alarm clearing events but other traps will simply provide information and not an alarm state.

The OSS/J QoS interface deliberately does not provide for alarms being cleared by specific users. The expectation is that when an alarm is cleared in the network, the interface will pass this cleared alarm to the network management system. The master source of alarm information is expected to be the elements in the network.

Therefore a basic requirement for an OSS/J interface on OpenNMS is to provide a means to examine SNMP traps and pair alarm raises with alarm clears to create an alarm list. Unfortunately, SNMP trap based management does not usually provide means to audit the current state of alarms in a network and clear events can be missed. Therefore it is possible for the alarm list to grow indefinitely if there were no manual means to remove alarms. In a full blown scenario, it would be necessary to create a more sophisticated mapping of multiple traps to specific alarms and also to allow for counts of received traps over a threshold window period to create or clear alarms and prevent alarm build up. This complex analysis is beyond the scope of the current project.

Fortunately, OpenNMS provides a simple event filtering capability within its existing event management system which can be used to ensure only events relevant to the scenario are presented to the correlation engine which will acknowledge those alarms once it creates a trouble ticket.

The OpenNMS team are embarking on a project to create an alarm list from received events which is expected to deliver a working solution in 2005. Unfortunately was too late for the current Catalyst project but we should consider planning to re-implement the demonstration interface to take advantage of this capability when it is available. This would be a relatively simple exercise which would provide a flexible OSS/J capability on OpenNMS.

Note that the OpenNMS OSS/J FM interface supports only the following operations;

a. Allow clients to register for alarm events and to establish an initial context for the OSS/J Qos interface
b. On request, return entire current alarm list with no filtering.
c. Any filters applied in a request should be ignored and still return the full list.
d. Note that the intention is to completely simplify the mechanism for retrieving the alarm list from OpenNMS to just returning the list and the alarm count values for the list

e. Acknowledge a set of alarms in the alarm list alarms given a set of alarms to acknowledge

When an SNMP event which is relevant to the scenario is received by OpenNMS it is treated as an alarm in the alarm list and a notification is raised on the OSS/J interface. Clients of the interface are then notified and can request the complete alarm list as an iterator. Note that for this simplified scenario no filtering has been provided on OpenNMS for the alarms provided. The entire list will is passed.

The correllator registers with the OpenNMS interface for alarms and updates its local list of alarms on each event. The rules in the correlator create a trouble ticket given a certain set of alarms. This set of alarms are then attached to the trouble ticket and automatically acknowledged back to the Alarm List. The alarm list in OpenNMS must then acknowledge all of the events which created this list of alarms.

G.6.6.3Correlation Engine

Figure 59 shows a Class diagram of Correlation solution. This is explained more fully in [4] but in outline the system works as follows;

The correlation engine uses the drools (www.drools.org) open source implementation of Forgy's Rete algorithm. It has a small but very active community continuously working to improve the engine and to help users write their own rules. Rules for drools are very easy to get to grips with, being written in native java within an xml format that specifies the parameters, conditions and consequence. This makes for very manageable and rapid development of rules. The drools engine also conforms to the jsr94 standard for incorporating inference engines in java programs. The jsr94 API allows loading and invocation of a rule set for a compliant rules engine both remotely and locally. The advantage of using the API means that the rules engine for a system can easily be interchanged at the cost of rewriting all the rules for the new engine. This means that it should be relatively easy to swap Drools for another inference engine supporting the same interface.

The solution runs within a JBoss container and Control bean is responsible for running the solution within JBoss. On start-up the rules engine loads the rules_file.drl which controls how it correlates related alarms.

The Alarm_event_beans_handler receives OSS/J event notifications that a change has occurred in the alarm list. When an event is received, a timer is started during which the system waits for other events to arrive which could correlate to the initial event. The system then uses the Qos Store handler to access the full alarm list through the OSS/J interface. This list is passed to the rule engine which determines whether a trouble ticket should be created from the list of alarms. The Ticket Collector then creates a trouble ticket which is passed through the OSS/J TT interface to OTRS. If a ticket is created, all of the alarms which are tagged to the ticket by the correlation engine are acknowledged through the OSS/J QoS interface.

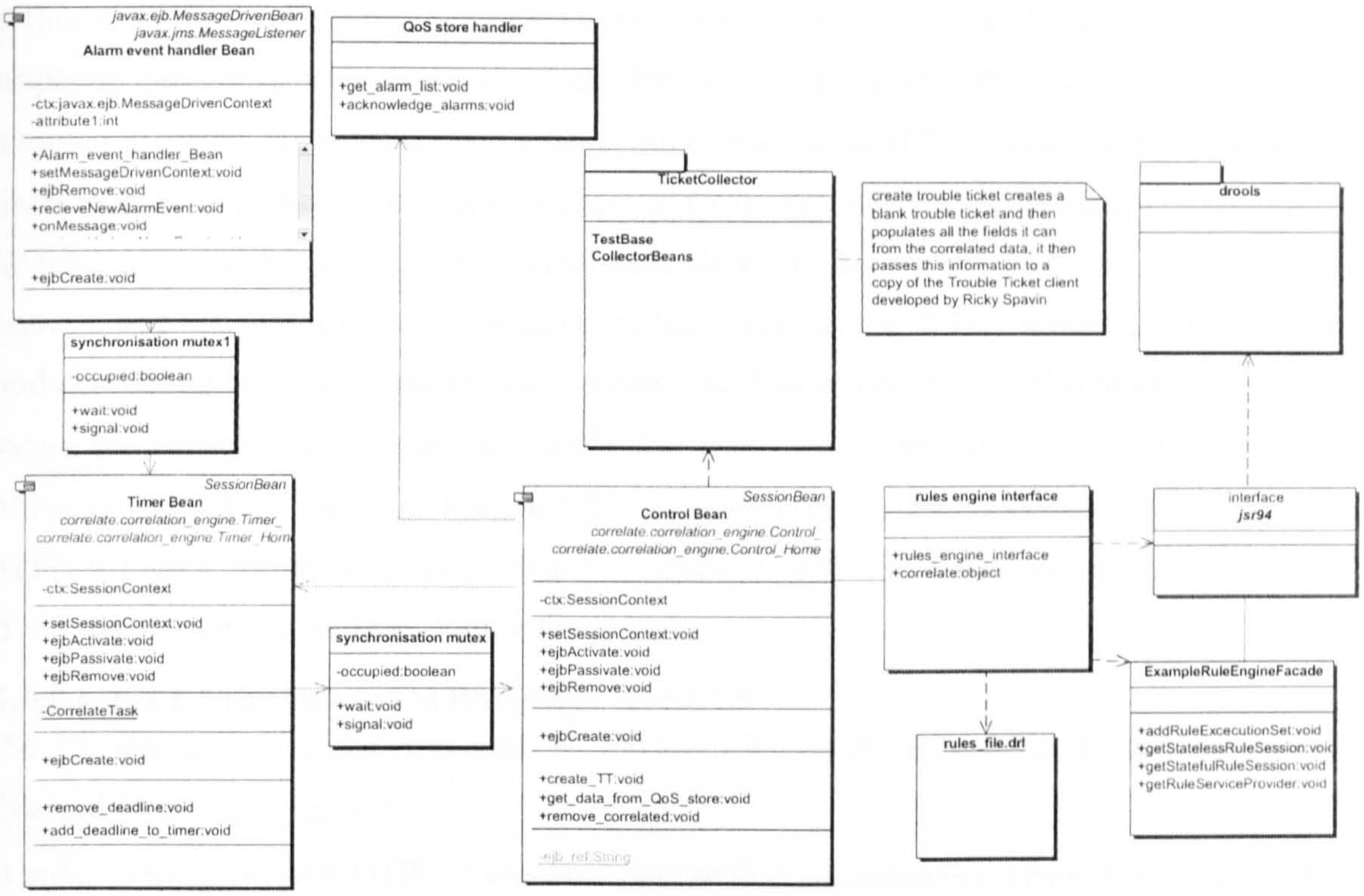


Figure 59 Class diagram of Correlation solution

G.6.6.4OSS/J TT interface to OTRS

In this section we discuss how an OSS/J TT interface was created for OTRS.

G.6.6.4.1 Overview of OTRS internal design

OTRS is built according to the MVC (Model View Controller) design paradigm. This means that the controlling code is separated from the user interface and the raw data. This structure allows for easy modification of each level of the system as there should be minimal coupling between the three sections. Figure 60 shows the basic architectural structure of the OTRS system, with the view model situated at the top, the controlling models in the middle and the data model at the bottom of the diagram.

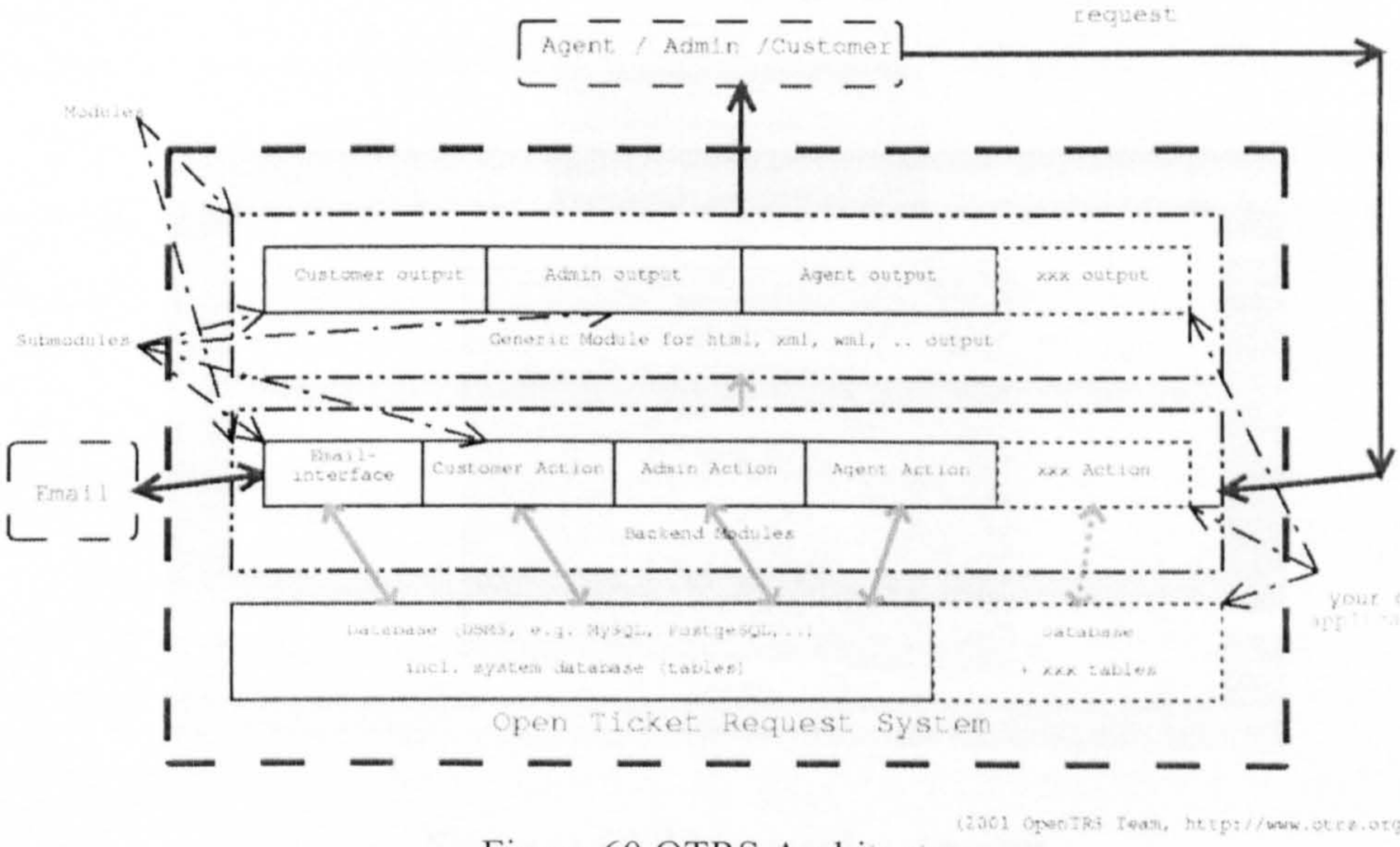


Figure 60 OTRS Architecture

In this scenario the view consists of a series of Perl modules that can be interpreted as CGI scripts to provide a web based interface that is used to trigger the underlying controller modules. Because the interface is via CGI (rather than plain HTML) the interface can make direct module calls inside the system to allow an easier event driven framework. The controller modules are used to provide the external data flows of the system, and also to process all requests and actions of the user. Because of the nature of the OTRS system many of these modules are used to provide interfaces to control the flow of emails, and also to allow database access for storage. All data storage inside the model, takes the form of an SQL database (although MySQL is standard, PostgreSQL can also be used). The database schema used by OTRS is highly normalised, using around 50 database tables, to ensure the minimum amount of redundancy and dependency within the system.

G.6.6.4.2 TT Interface to OTRS Design overview

The TT interface to OTRS was built by reusing parts of the OSS/J TT RI implementation obtained from www.ossj.org

In order to integrate with OTRS it was necessary to find a method that allowed messages to be passed directly from the OSS/J bus to the system with minimal alterations to OTRS itself. Avoid modification to OTRS wherever possible, minimised the debugging process, and increase the development speed.

It was decided that the most effective implementation would require three basic layers. One to interact directly with the information received by the OSS/J bus, one to directly access the OTRS database, and an intermediary layer that will perform all the necessary conversions to allow seamless interaction between the components. These three components were then wrapped up in a bean and deployed on the same server as the modified reference implementation. In order to simplify the interaction between the sections of the program all of the integration code is java. This allows simple object calls and methods to be used, as opposed to indirect invocation by an alternative language.

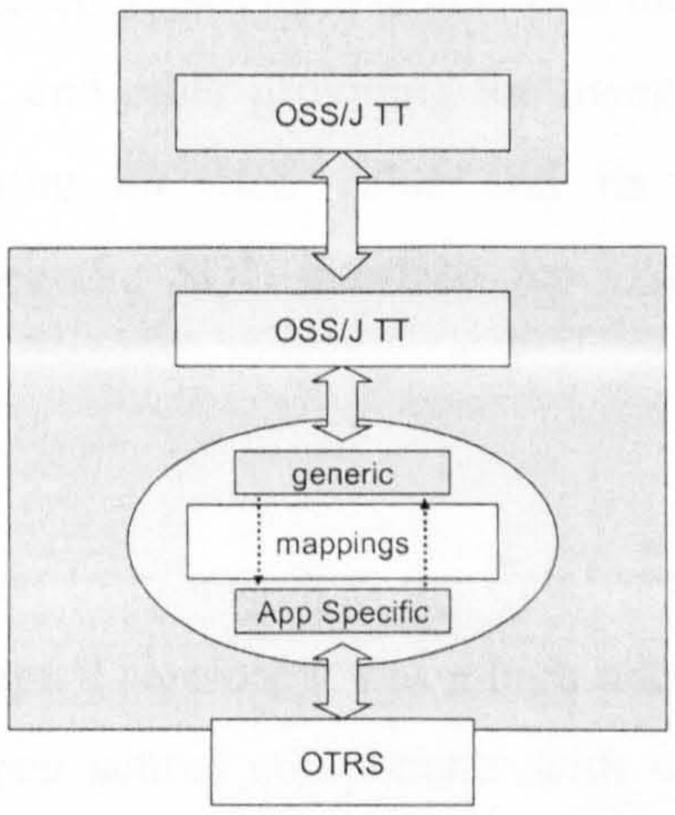


Figure 61 OTRS integration diagram

The layer referred to as "Generic" in Figure 61, is the layer that directly interacts with the TT reference implementation. The interfaces provided by the generic section are used to recognise, accept, and process the necessary operations as defined by the OSS/J framework. This set of functions will be required by any compliant OSS/J trouble ticketing system, and acts as the basis of the integration with OTRS. This section should never require modification regardless of the trouble ticketing system that is used. When the generic methods are invoked, the necessary SQL strings will be generated by the Mappings layer, before being executed in the application specific section.

The application side of the bean is comprised of a layer that is used to actually perform the operations that form the interaction between the trouble ticketing system and the OSS/J components. In this situation the functions will be limited to creating a connection to the database and perform the necessary updates and queries. These functions have been segregated as being application specific, as the database connection and the method of invocation may vary between trouble ticketing solutions. Currently these operations are performed using a JDBC connector that allows the Java classes to interact with the database in the same manner as a user would interact with the database through the database client. The operations to be processed are passed as Strings from the Generic layer, and can be directly executed with any corresponding return sent directly to the calling method. As this section requires explicit knowledge of the trouble ticketing system, this section of the bean will need to be changed for each different type of database or application that is used.

Sitting in between these two components is the "Mappings" layer. This is where the variables stored in the OSS/J trouble ticket that can be handled by the trouble ticketing system need to be defined, along with details of the underlying database schema that defines where the values should be stored. The structure of this layer has been specifically designed so all differences between systems can be resolved using configuration files rather than directly altering the source code. The details of the variables and the database schema are loaded in from file, and the necessary conversions are stored in two hash tables; One mapping the OSS/J variable name to the database table and field, and other providing the inverse correlation between variable and database entry. By looking up each value and its final destination before each modification, a non-schema specific SQL interface for OSS/J trouble ticketing has been created.

G.7 Experiences

It is important to understand that from inception the Catalyst project had 3 months to design and integrate the proof of concept. Therefore it was a high pressure project which unlike other Catalyst projects was using open source components with which none of the parties were highly familiar. In future phases of this project, we hope to have increased our familiarity with

the open source components available and be able to concentrate more on defining the service scenarios.

In addition to raising the profile of the OpenOSS project, the catalyst provided the following proof points for the project concept.

Firstly, we proved that suitable open source components exist at a level of maturity for us to proceed with this project. Furthermore the communities surrounding these projects appear to be accessible and helpful. In particular both the OpenNMS and Drools communities provided us with helpful support.

The disparate companies were able to work together using the open source approach. However the management style had to recognise that the resources were being provided as good will – there were no formal contracts binding us to deliverables. This required flexibility in the definition of the solution to match what each party felt they could contribute.

As a team, we felt we had a lack of experience of the open source development cycle – and this is an area we will wish to develop quickly through our engagements with more established open source projects. In particular, we will need to think through our code management process and decide which development tools are most appropriate going forwards.

The experience of building the proof of concept provided an excellent vehicle for beginning to appreciate how OSS/J and NGOSS can be practically applied. We believe that further work will deliver some really useful experience to the industry. We believe that NGOSS provides a very useful framework for understanding an OSS architecture and we hope to utilise it more fully going forwards.

OSS/J itself is functionally rich but lacking in tool support. Since the Catalyst, we have discovered an emerging project which has a mixed commercial / open source business model (*TigerStripe*) seeking to address the OSS/J tooling problem and hope to work with them in the future.

Participating in a catalyst provided a very intense and rapid vehicle for getting a lot of work done collaboratively in a short time for a proof of concept. However, to do so, we did cut a lot of corners and would like to take more time about defining the next stages of the project. However the biggest benefit of the Catalyst was the visibility it gave us across the industry – and the potential contacts to build a sustainable open source community going forwards.

G.8 Next Steps

Having just completed our Catalyst demonstration in May 2005, our first priority will be to document our work and make the code publicly available as an early proof of concept.

We are in the process of formulating plans for a longer term project which will seek to establish an open source community under the auspices of the OpenOSS Initiative. We are actively seeking partnership across the OSS business community and within the Open Source and Academic communities in order to realise the vision of OpenOSS.

G.9 Conclusions

The approach of OpenOSS has been welcomed by the Telemanagement Forum as a means to involve more academic partners in their activities and also provide a 'living lab' to demonstrate NGOSS principles. The basic principles of this way of working have been tested through the initial 'Catalyst' demonstration and in parallel a program of research and development is being developed which the partners are currently discussing converting into a project consortium.

G.10 References

Reference	Description	Use by the Project
[1]	Open OSS Project Charter Version 1.0 Dec 2004	Sets Project business scope, objectives and deliverables
[2]	Open OSS Project Initiation Document 0.2 Jan 2004	Documents project structure, plan, deliverables, dependencies and risks
[3]	Extension of OpenNMS to NGOSS Architecture via OSS/J	Gavin Willingham (University of Southampton student) project report
[4]	Using the NGOSS Lifecycle with an MDA approach to provide a re-usable and maintainable problem correlation solution	Tim Griffith (University of Southampton student) project report

G.11 Document History

Revision	Date	Author	Description
0.2	03-06-05	Craig Gallen	First draft
0.2b		Doug Carson	Updated with more Agilent information
0.3	04-7-05	Craig Gallen	Updated with comments

This page is deliberately left blank